

Untersuchung eines Verfahrens zur Stichprobenziehung auf Graphen

Peter Neubert

Geboren am 19. Juli 1989 in Seeheim-Jugenheim

01.09.2016

Bachelorarbeit Mathematik

Betreuer: Prof. Dr. Jochen Garcke

Zweitgutachter: Dr. Tino Ulrich

INSTITUT FÜR NUMERISCHE SIMULATION

Betreuer am Fraunhofer Institut: Victor Rodrigo Iza-Teran

FRAUNHOFER-INSTITUT FÜR ALGORITHMEN UND WISSENSCHAFTLICHES RECHNEN

MATHEMATISCH-NATURWISSENSCHAFTLICHE FAKULTÄT DER
RHEINISCHEN FRIEDRICH-WILHELMS-UNIVERSITÄT BONN

Inhaltsverzeichnis

1	Einleitung	1
2	Signalverarbeitung auf Graphen	5
2.1	Signale auf Graphen	5
2.2	Abtastung und Rekonstruktion von Signalen	8
2.3	Filter auf Graphen	12
2.3.1	Konstruktion von Filtern	12
2.3.2	Polynomielle Approximation eines Filters	14
3	Das betrachtete Verfahren	17
3.1	Das Abtasten der Signale	17
3.2	Approximation der optimalen Verteilung \mathbf{p}^*	21
3.2.1	Schätzung der Einträge von \mathbf{p}^*	22
3.2.2	Schätzung des Eigenwertes λ_k	24
3.3	Rekonstruktion k -bandlimitierter Signale	26
3.4	Implementierung des Algorithmus	30
4	Numerische Experimente	33
4.1	Untersuchung der synthetischen Daten	33
4.1.1	Der Regularisierungsterm $\gamma z^\top g(L)z$	34
4.1.2	Die Anzahl	36
4.1.3	Der Einfluss von k und eps	38
4.2	Automobildaten im Fraunhofer SCAI	39
4.3	Untersuchung des Bauteils 2000002	42
5	Abschließende Betrachtungen	51
	Notations- und Abkürzungsverzeichnis	55
	Abbildungsverzeichnis	57
	Literaturverzeichnis	59

1 Einleitung

Die Verarbeitung großer Datenmengen spielt sowohl in der numerischen Simulation, als auch in der Industrie eine immer größer werdende Rolle. Die Daten sind dabei oft in einen strukturellen Kontext eingebettet, wovon eine mögliche Struktur ein Graph ist. Ein Modell welches sich besonders zur Modellierung von Netzwerken jeglicher Art anbietet. Durch die Wahl eines solchen Modells kann die zugrundeliegende Struktur mit in den Analyseprozess einbezogen werden; das Verhalten der Daten im Kontext der zugrundeliegenden Geometrie und Topologie betrachtet werden [3].

Dabei stellt sich die Frage wie die Graphenstruktur in den Analyseprozess gewinnbringend eingebunden werden kann.

Wenn es die Situation zulässt, können die Daten als skalare Werte auf den Knoten interpretiert werden. Dadurch kann der Datensatz als Vektor aufgefasst werden, wobei der Eintrag an der Stelle i dem Wert entspricht, der auf dem i -ten Knoten des Graphen angenommen wird. Die Länge des Vektors ist dann durch die Anzahl der Daten gegeben. Der Vektor v kann als diskretes Signal auf dem Graphen gesehen werden, wodurch sich die Frage ergibt, ob die Methoden der Signalverarbeitung auf Graphen zur Unterstützung der Datenanalyse genutzt werden können.

Innerhalb der Signalverarbeitung nimmt das Abtasten eines Signales eine zentrale Rolle ein. Dabei wird ein Signal \mathbf{s} zu verschiedenen Zeitpunkten t_1, \dots, t_m gemessen und die gemessenen Signalwerte $\mathbf{s}_1, \dots, \mathbf{s}_m$ gespeichert. Dadurch entsteht ein neues Signal mit den Einträgen $(\mathbf{s}_1, \dots, \mathbf{s}_m)$ [1]. Es kann als Signal in einem neuem Raum aufgefasst werden, wobei dieser Raum eine entsprechend kleinere Dimension hat.

Dabei sollen die Messungen möglichst so vorgenommen werden, dass das ursprüngliche Signal daraus rekonstruiert werden kann. Dabei stellt sich vor allem der Einsatz der Fourier-Transformation als zentrales Hilfsmittel heraus. Dabei kommt der Laplace-Operator zum Einsatz [5].

Innerhalb der Signalverarbeitung auf Graphen wird die Fourier-Transformation mit Hilfe der Laplace-Matrix L realisiert. Dabei wird L mit Hilfe der Adjazenzmatrix des Graphen gebildet und bezieht somit die Struktur des Graphen mit ein. Die Menge U die aus den Eigenvektoren der Matrix L besteht kann dann zur Realisierung der Transformation benutzt werden. U ist somit eine Fourier-Basis, bezüglich derer die Signale auf dem Graphen dargestellt werden können [5].

Ein Signal auf einem Graphen wird als k -bandlimitiert bezeichnet, wenn nur die ersten k Fourier-Koeffizienten nicht null sind. k -bandlimitierte Signale lassen eine perfekte Rekonstruktion zu; das ursprüngliche Signal kann aus den Messungen exakt wiederhergestellt werden. Daher nehmen diese Signale einen besonderen Platz ein, da eine perfekte

Rekonstruktion ist im Allgemeinen nicht zu ist.

Es gibt verschiedene Arten, um Messungen an Signalen vorzunehmen. Genannt seien hier *randomisiert*, dabei werden die Messungen zu zufälligen Zeitpunkten vorgenommen.

Und *deterministisch*, die Messungen werden an klar bestimmten Zeitpunkten vorgenommen. Bestandteil dieser Arbeit sind randomisierte Messungen von bandlimitierten Signalen.

Problemstellung

Die k -bandlimitierten Signale müssen in einen Raum mit niedriger Dimension eingebettet werden. Dazu muss ein geeignetes Verfahren zur Abtastung konstruiert und es müssen genügend Messungen vorgenommen werden. Beides ist möglich und kann zum Beispiel mit den Eigenvektoren der Laplace-Matrix umgesetzt werden. Der Aufwand zur Berechnung der Vektoren ist dabei in $\mathcal{O}(n^3)$, wobei n die Knotenzahl ist [11]. Hierbei kommt ein bekanntes Problem zum tragen. Der Aufwand, der für die Zerlegung von L betrieben werden muss, ist für große Graphen schlicht zu hoch.

An dieser Stelle setzt der Artikel *Random sampling of bandlimited signals on graphs* [1] der Autoren *G. Puy, N. Tremblay, R. Gribonval, P. Vandergheynst* an. Darin werden im wesentlichen zwei Dinge präsentiert. Die Autoren konstruieren ein Abtastverfahren für bandlimitierte Signale, das ohne die Zerlegung der Laplace-Matrix auskommt. Im Anschluss stellen sie ein Verfahren zur Rekonstruktion dieser Signale bereit.

Aufbau und Ziele der Arbeit

Die vorliegende Arbeit verfolgt zwei Hauptziele. Zum einen die Implementierung der in [1] vorgestellten Ideen sowie deren genauere Untersuchung und zum anderen die Anwendung der Implementierung auf Simulationsdaten. Bei den Daten handelt es sich um simulierte Crashtests. Die Datensätze sind durch Finite-Elemente Gitter gegeben aus denen die benötigten Informationen und der zu Grunde liegende Graph extrahiert werden kann.

Hierbei steht weniger eine konkrete Zielsetzung im Raum, als viel mehr die Untersuchung inwiefern der Einsatz des Verfahrens auf den Daten etwaige Vorteile verspricht. Dabei sollen die xyz -Koordinaten darauf getestet werden, ob sie als k -bandlimitiertes Signal betrachtet werden können für ein $k \in \mathbb{N}$. Darauf aufbauend ergibt sich die Frage, ob die anschließende Rekonstruktion gut genug ist, um einen Ansatz der Datenkompression darzustellen. In diesem Fall sollte die Rekonstruktion entsprechend genau sein. Des Weiteren empfand ich die Darstellung der Thematik in [1] als unvoreteilhaft. An einigen Stellen hatte ich Schwierigkeiten der Argumentation der Autoren zu folgen. Daher sah ich es als Teil meiner Arbeit an besagten Artikel übersichtlicher aufzuarbeiten.

Zum Aufbau der Arbeit:

Kapitel 2 stellt eine kurze Einführung in die Signalverarbeitung auf Graphen dar. Dabei wurden die Themen so gewählt, dass sie das für das Verständnis des Algorithmus

in [1], benötigte Theoriegebäude aufbauen. Im 3. Kapitel wird das Vorgehen der Autoren genauer betrachtet und strukturiert aufgearbeitet. Die theoretischen Ergebnisse und die algorithmische Umsetzung werden hergeleitet und implementiert. Kapitel 4 beschreibt die Ergebnisse der numerischen Testreihen auf den synthetischen Daten und auf Datensätzen der Automobilindustrie.

Eigene Leistung

Während der Entstehung dieser Arbeit orientierte sich mein Vorgehen vor allem um die folgenden Ziele.

- Einarbeitung in die Themen *Signalverarbeitung auf Graphen* und *Compressive Sensing*.
- Implementierung der in [1] vorgestellten Verfahren, sowie die Reproduktion der Ergebnisse.
- Analyse der Parameter und deren Einfluss auf das Verfahren. Ferner eine genauere Betrachtung von zwei Ansätzen zur Verbesserung der Ergebnisse.
- Untersuchung von Crash-Test Daten im Hinblick auf die sinngebende Anwendung der Implementierungen.

Danksagung

Meiner Verlobten Mirijam Franziska Gütersloh möchte ich für die Unterstützung nicht nur während der Entstehung dieser Arbeit, sondern auch für die Unterstützung während des gesamten Studiums danken. Ohne deine Hilfe hätte ich es nicht geschafft. Ebenso danke ich meiner Familie, die mich stets unterstützte.

Jochen Garcke und Rodrigo Iza-Teran danke ich für die umfassende Betreuung bei sämtlichen mathematischen und inhaltlichen Fragen sowie der enormen Hilfe bei der Abwicklung von organisatorischen Dingen die Universität betreffend. Mandar Satyanath Pathare danke ich für die Unterstützung und Hilfe in sämtlichen Fragen rund um das Thema Software und Programmieren und für all die Tipps und Erfahrungen die er mit mir geteilt hat. Letztlich bedanke ich mich bei allen Mitarbeitern des Fraunhofer SCAI für die vielen Anregungen und Ideen, welche zur Fertigstellung der vorliegenden Arbeit beitrugen.

Pierre Vandergheynst und insbesondere auch Gilles Puy danke ich für die Beantwortung der Fragen bezüglich der Implementierung des Algorithmus und für die Bereitstellung ihres MATLAB-Codes [19]. Dadurch konnten viele offene Fragen beantwortet werden.

Letztlich danke ich allen Kommilitonen und Freunden, die mir mit Rat und Tat zur Seite standen.

2 Signalverarbeitung auf Graphen

Im folgen möchte ich das für diese Arbeit benötigte Theoriegebäude aufbauen und darstellen. Hierfür soll im ersten Abschnitt die benötigten Grundlagen eingeführt werden. Danach sollen die beiden Gebiete *Abtastung und Rekonstruktion* sowie *Filter auf Graphen* in ihren Grundzügen dargestellt werden. Der letzte Abschnitt beschreibt die Approximation eines Filters auf Basis eines Polynoms.

Die Hauptquellen die ich zur Erstellung für diesen Abschnitt nutzte waren [1], [2], [3], [5] für die Themen zur Signalverarbeitung und [10] für die Einführung zur Abtastung und zur Rekonstruktion. Vor allem der Artikel [5] liefert einen guten ersten Überblick.

2.1 Signale auf Graphen

Im Folgenden sei $G = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ stets ein ungerichteter, zusammenhängender Graph mit endlicher Knotenmenge $\mathcal{V} = \{v_1, \dots, v_n\}$, Kantenmenge \mathcal{E} und Adjazenzmatrix $\mathcal{W} \in \mathbb{R}^{n \times n}$. Dabei seien die Einträge von \mathcal{W} positiv.

Definition 2.1.1 *Sei G ein Graph. Ein **Signal** \mathbf{s} auf G ist eine Abbildung*

$$\begin{aligned} \mathbf{s} : \mathcal{V} &\longrightarrow \mathbb{R} \\ v_i &\longmapsto \mathbf{s}_i. \end{aligned}$$

Da die Knoten von G aufgrund der Indexmenge mit einer Reihenfolge versehen sind und somit eindeutig durchlaufen werden, kann ein Signal auch als Vektor

$$\mathbf{s} = (\mathbf{s}_1, \dots, \mathbf{s}_n)^\top \in \mathbb{R}^n$$

aufgefasst werden [3], wobei der i -te Eintrag \mathbf{s}_i als der i -te Signal-Wert aufgefasst wird. Alle Signale die im Rahmen dieser Arbeit eine Rolle spielen sind als Vektoren gegeben. Zur besseren Anschauung zeigt Abbildung 2.1 eine Beispiel eines Signals auf einem Netzwerk.

Für einen gegebenen Graphen G mit n Knoten bezeichne \mathcal{S}_G die Menge der Signale auf G . Jedes dieser Signale kann als Vektor in \mathbb{R}^n aufgefasst werden und folglich kann auch jeder gegebene Vektor aus \mathbb{R}^n als Signal auf G interpretiert werden [3]. Daher gilt

$$\mathcal{S}_G = \mathbb{R}^n.$$

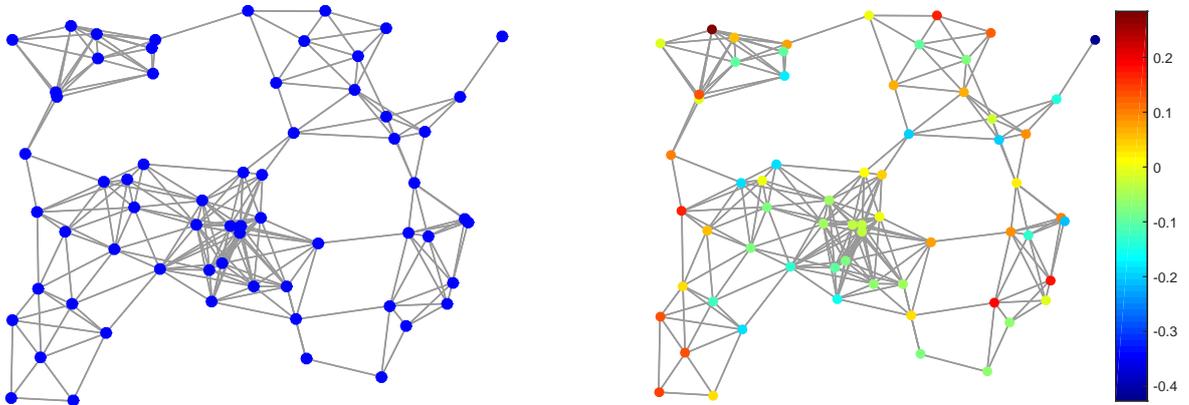


Abbildung 2.1: Beispiel eines Signals auf einem Graphen. Links der ursprüngliche Graph, rechts ein zufälliges Signal darauf. Die unterschiedlichen Farben repräsentieren dabei unterschiedliche Signalwerte [16].

Daher möchte ich die Notation \mathcal{S}_G insbesondere dann nutzen, wenn Signale auf einem bestimmten Graphen untersucht werden. Die Notation \mathbb{R}^n soll hingegen ausdrücken, dass ein beliebiger Graph untersucht wird.

Definition 2.1.2 Für einen gegebenen Graphen G ist die (kombinatorische-) **Laplace Matrix** gegeben durch

$$L := \mathcal{D} - \mathcal{W}. \quad (2.1)$$

Dabei bezeichnet $\mathcal{D} = (\mathcal{D})_{ij}$ die **Gradmatrix** mit den Einträgen $\mathcal{D}_{ii} := \sum_{i \neq j} \mathcal{W}_{ij}$. In Anlehnung an die diskrete Signalverarbeitung werden die Eigenwerte λ_i von L auch als **Frequenzen** bezeichnet.

In der Literatur ist auch der Begriff *unnormalisierte Laplace-Matrix* geläufig. Da in der vorliegenden Arbeit nur die kombinatorische Laplace-Matrix von Interesse ist, werde ich das Wort *kombinatorisch* nicht länger erwähnen.

Eine zentrale Eigenschaft der Laplace-Matrix ist hierbei von besonderen Interesse.

Satz 2.1.1 Die Laplace-Matrix ist eine reelle, symmetrische und positiv semi-definite Matrix.

Beweis: An dieser Stelle folge ich dem Beweis aus [13].

1. Da die Adjazenzmatrix \mathcal{W} symmetrisch ist für ungerichtete Graphen und \mathcal{D} eine Diagonalmatrix ist, folgt die Symmetrie von L . Da sowohl \mathcal{D} als auch \mathcal{W} reell überträgt sich diese Eigenschaft auf L .

2. Sei $x = (x_1, \dots, x_n)^\top \in \mathbb{R}^n \setminus \{0\}$. Dann gilt:

$$\begin{aligned}
 0 &\leq \sum_{i,j=1}^n (x_i - x_j)^2 \mathcal{W}_{ij} = \sum_{i,j=1}^n (x_i^2 + x_j^2 - 2x_i x_j) \mathcal{W}_{ij} \\
 &= \sum_{i=1}^n x_i^2 \mathcal{D}_{ii} + \sum_{j=1}^n x_j^2 \mathcal{D}_{jj} - 2 \sum_{i,j=1}^n x_i x_j \mathcal{W}_{ij} \\
 &= 2 \left(x^\top (\mathcal{D} - \mathcal{W}) x \right) = 2 \left(x^\top L x \right).
 \end{aligned}$$

■

Da L reell und symmetrisch ist existiert eine orthogonale Zerlegung

$$L = U \Lambda U^\top$$

wobei $\Lambda = \text{diag}[\lambda_1, \dots, \lambda_n]$ die Eigenwerte von L enthält und U die entsprechenden Eigenvektoren. Dabei sind alle Eigenwerte positiv [1].

Die Zerlegung liefert eine neue Basis U bezüglich derer die Signale dargestellt werden können. Die Signale auf G können somit auf zwei Arten dargestellt werden, was für diese Arbeit eine wichtige Rolle spielt.

- Ist ein Signal \mathbf{s} auf den Konten des Graphen G direkt angegeben so spricht man auch von einer Definition auf der *Knotendomäne*. Diese Darstellung entspricht der anschaulichen Vorstellung. Die Menge der Signale in der Knotendomäne entspricht dabei \mathcal{S}_G .
- Ist es dagegen entsprechend der Basis U dargestellt so ist \mathbf{s} in der *Spektraldomäne* gegeben. Analog bezeichne $\hat{\mathcal{S}}_G$ die Spektraldomäne von G .

Dabei existiert eine Transformation von der Knotendomäne in die Spektraldomäne und auch eine entsprechende inverse Abbildung, die Graph Fourier-Transformation.

Definition 2.1.3 Sei G ein Graph mit Laplace-Matrix L und entsprechender Zerlegung $L = U \Lambda U^\top$. Ferner sei $\mathbf{s} \in \mathcal{S}_G$ ein Signal auf G . Dann heißt die Abbildung

$$\begin{aligned}
 \text{GFT} : \mathcal{S}_G &\longrightarrow \hat{\mathcal{S}}_G \\
 \mathbf{s} &\longmapsto U^\top \mathbf{s} =: \hat{\mathbf{s}}
 \end{aligned} \tag{2.2}$$

Graph Fourier-Transformation. Die Einträge von $\hat{\mathbf{s}}$ heißen **Fourier-Koeffizienten** und beschreiben den Gehalt der Frequenzen von L in aufsteigender Reihenfolge.

Die entsprechende inverse Abbildung

$$\begin{aligned}
 \text{IGFT} : \hat{\mathcal{S}}_G &\longrightarrow \mathcal{S}_G \\
 \hat{\mathbf{s}} &\longmapsto U \hat{\mathbf{s}} = \mathbf{s}
 \end{aligned} \tag{2.3}$$

heißt **inverse Graph Fourier-Transformation**.

Durch Übergang zur Spektraldomäne offenbaren sich demnach die dem Signal \mathbf{s} zugrundeliegenden Frequenzen und deren Gewichte. Da in dieser Arbeit nur eine Fourier-Transformation von Bedeutung ist, werde ich das Wort *Graph* weglassen und nur von Fourier-Transformation sprechen.

Die Signale, welche Gegenstand der vorliegenden Arbeit sind bilden eine eigene Klasse von Signalen. Die k -bandlimitierten Signale. Ihre charakterisierende Eigenschaft liegt darin begründet, dass sie nur von den ersten k Basisvektoren von U abhängen [1].

Definition 2.1.4 Sei G ein Graph mit Laplace-Matrix $L = U\Lambda U^\top$. Ein Signal \mathbf{s} auf G heißt **k -bandlimitiert**, wenn es ein $k \in \mathbb{N} \setminus \{0\}$ gibt mit

$$\mathbf{s} \in \text{span} [u_1, \dots, u_k]. \quad (2.4)$$

Alternativ zur Definition 2.4 können besagte Signale in der Spektraldomäne definiert werden [2]. Hier ergibt sich eine ähnliche Definition.

Definition 2.1.5 Ein Signal \mathbf{s} auf G heißt **k -bandlimitiert**, wenn es ein $k \in \mathbb{N} \setminus \{0\}$ gibt, sodass für die Fourier-Koeffizienten von $\hat{\mathbf{s}} = U^\top \mathbf{s}$ gilt:

$$\hat{\mathbf{s}}_i = 0 \quad \forall i > k. \quad (2.5)$$

Bemerkung: Das in Abbildung 2.1 gezeigte Signal ist ein k -bandlimitiertes Signal mit $k = 50$.

2.2 Abtastung und Rekonstruktion von Signalen

Beim Abtasten eines Signals $\mathbf{s} = (\mathbf{s}_1, \dots, \mathbf{s}_n)^\top \in \mathbb{R}^n$ werden aus den n Einträgen m Stück ausgewählt, wobei $m < n$. Dadurch entsteht ein neues Signal in \mathbb{R}^m . Besagter Vorgang kann durch die Multiplikation mit einer Abtastmatrix $M \in \mathbb{R}^{m \times n}$ realisiert werden.

$$\begin{aligned} M : \mathbb{R}^n &\longrightarrow \mathbb{R}^m; \\ \mathbf{s} &\longmapsto M\mathbf{s} =: \mathbf{s}_M \end{aligned} \quad (2.6)$$

Die gegensätzliche Operation, bei der aus einem reduzierten Signal \mathbf{s}_M das ursprüngliche wieder gewonnen werden soll, wird als Rekonstruktion bezeichnet. Auch dieser Vorgang kann mittels einer Multiplikation mit einer Matrix $\Phi \in \mathbb{R}^{n \times m}$ realisiert werden.

$$\begin{aligned} \Phi : \mathbb{R}^n &\longrightarrow \mathbb{R}^m; \\ \mathbf{s}_M &\longmapsto \Phi \mathbf{s} =: \mathbf{s}' \end{aligned} \tag{2.7}$$

Es stellt sich die Frage, wie gut die Rekonstruktion \mathbf{s}' an \mathbf{s} ist, ob zum Beispiel der Fall $\mathbf{s}' = \mathbf{s}$ erreicht werden kann. Die Antwort darauf lautet allerdings nein, nicht im Allgemeinen, denn es gilt:

$$\mathbf{s}' = \Phi \mathbf{s}_M = \Phi M \mathbf{s} \Leftrightarrow \Phi M = \mathbf{I}_n, \tag{2.8}$$

wobei $\text{rank}(\Phi M) \leq k < n = \text{rank}(\mathbf{I}_n)$ und somit $\Phi M \neq \mathbf{I}_n$. Perfekte Rekonstruktion ist also im Allgemeinen nicht möglich [2].

Bevor ich weiter auf die Thematik eingehe möchte ich kurz anreißen, wo das Abtasten und das Rekonstruieren von Signalen in der Praxis Anwendung finden kann:

1. Sensornetzwerk

Gegeben sei ein Netzwerk bestehend aus n Sensoren, die alle stets zum selben Zeitpunkt eine Messung vornehmen. Das zugrundeliegende Netzwerk sei ein zusammenhängenden Graph. Die Signale auf diesem Netzwerk sind dann gerade die n Messungen zu einem festen Zeitpunkt t .

Jetzt seien k der n Sensoren defekt. Die Signale bestehen jetzt nur noch aus $(n - k)$ Messungen, die Messwerte an den defekten Stellen sind nicht verfügbar. Kann das Signal allerdings rekonstruiert werden, so könnten die Messwerte an den defekten Stellen rekonstruiert werden.

2. Datenkompression

Angenommen aus einem gesampelten Signal \mathbf{s}_M kann das ursprüngliche Signal genügend gut rekonstruiert werden. Dann kann der Speicherbedarf für das Signal \mathbf{s} auf den von \mathbf{s}_M reduziert werden. Wenn jetzt die Zahl der gespeicherten Signale sehr groß ist und/ oder $m \ll n$ kann eine entsprechend große Menge an Speicher gespart werden.

Definition 2.2.1 Sei $\mathbf{s} \in \mathbb{R}^n$ ein Signal.

(i) Definiere in Anlehnung zur p -Norm $\|\mathbf{s}\|_0 := \#\{i \in \mathbb{N} \mid \mathbf{s}_i \neq 0\}$ also die Anzahl der nicht-null Einträge in \mathbf{s} .

Definiere ferner die Menge $\Sigma_k := \{\mathbf{s} \in \mathbb{R}^n \mid \|\mathbf{s}\|_0 \leq k\}$

(ii) Ein Signal \mathbf{s} heißt **k -dünnbesetzt** wenn $\mathbf{s} \in \Sigma_k$ gilt.

Wie in (2.8) gesehen scheitert der Versuch einer perfekten Rekonstruktion im Allgemeinen an dem unterbestimmten Gleichungssystem $\Phi \mathbf{s}_M$. Wenn jetzt allerdings \mathbf{s} dünnbesetzt ist kann die Rekonstruktion trotzdem gelingen. Gemäß Definition 2.5 können k -bandlimitierte Signale in der Spektraldomäne auch k -dünnbesetzte Vektoren aufgefasst werden. Demnach besteht bei bandlimitierten Signalen die Möglichkeit der perfekten Rekonstruktion.

Der Themenkomplex, der sich mit der Approximation durch möglichst dünnbesetzte Vektoren befasst, wird *Compressive Sensing* genannt.

Bemerkung: *Das Gebiet COMPRESSIVE SENSING ist ein sehr junges Forschungsgebiet, deren Fachartikel fast alle auf englisch publiziert werden. Für viele Fachbegriffe existiert noch keine deutsche Übersetzung, die sich in den Artikeln wiederfindet. Daher habe ich die wichtigen Fachbegriffe nicht übersetzt, sondern im englischen Original gelassen.*

Ein erste Ansatz um eine Rekonstruktion für \mathbf{s} zu bestimmen ist durch die Lösung des L_0 - Minimierungsproblems gegeben [10]:

$$\min_{z \in \mathbb{R}^n} \|z\|_0 \quad \text{u.d.N.} \quad Mz = \mathbf{s}_M \quad (2.9)$$

Der Ansatz zur Rekonstruktion besteht hierbei darin z als Rekonstruktion von \mathbf{s} zu akzeptieren, wenn die Abgetasteten Versionen z_M und \mathbf{s}_M übereinstimmen und z gleichzeitig eine minimale Anzahl an Einträgen besitzt.

Das Lösen von (2.9) ist im Allgemeinen allerdings NP-schwer [10], was eine algorithmische Lösung schnell zu aufwendig werden lässt.

Um dieses Problem zu umgehen, wird ein ähnliches Problem formuliert [10].

$$\min_{z \in \mathbb{R}^n} \|z\|_1 \quad \text{u.d.N.} \quad Mz = \mathbf{s}_M \quad (2.10)$$

Dabei handelt es sich um ein konvexes Optimierungsproblem, welches durch *Lineare Programmierung* gelöst werden kann. Die algorithmische Lösbarkeit von (2.10) ist damit wesentlich besser. Diese Vorgehensweise beruht auf der Hoffnung, dass eine Lösung von (2.10) eine vertretbare Lösung von (2.9) darstellt. Das ist der Fall wenn $\|z\|_0 \leq k$ und $\|z\|_1 \leq k$ gilt.

Bemerkung: *Analog zum Fall $p = 1$ in (2.10) kann auch jeder Wert $p > 1$ gewählt werden.*

Definition 2.2.2 (NSP-Eigenschaft) Eine Matrix $M \in \mathbb{R}^{m \times n}$ erfüllt die **Null-Space-Property** NSP der Ordnung k mit Konstante $\gamma \in (0, 1)$ wenn gilt:

$$\|v_T\| \leq \gamma \|v_{T^c}\|$$

für alle $T \subseteq \{1, \dots, n\}$ mit $|T| \leq k$ und für alle v aus $\ker(M)$. Dabei bezeichnet $T^c := \{1, \dots, n\} \setminus T$ das Komplement von T .

Als nächstes möchte ich darstellen, dass die (NSP)-Eigenschaft eine gute Matrix zur Abtastung auszeichnet.

Satz 2.2.1 Sei $\mathbf{s} \in \mathbb{R}^n$ ein Signal und $M \in \mathbb{R}^{m \times n}$ eine Abtastmatrix mit der NSP-Eigenschaft der Ordnung k für einen geeigneten Werte von γ . Ferner sei x^* die Rekonstruktion an \mathbf{s} , welche durch Lösen von (2.10) hervorgeht. Dann gilt:

$$\|\mathbf{s} - x^*\|_1 \leq \frac{2(1 + \gamma)}{1 - \gamma} \inf_{z \in \Sigma_k} \|x - z\|_1.$$

Insbesondere folgt für $\mathbf{s} \in \Sigma_k$, dass

$$\|\mathbf{s} - x^*\|_1 = 0 \iff x = x^*.$$

Beweis: Siehe [10], Theorem 3.2. ■

Wie schon oben erwähnt ist damit die Möglichkeit einer perfekten Rekonstruktion von einigen k -bandlimitierten Signalen in der Spektraldomäne gezeigt.

Allerdings ist es sehr komplex für eine gegebene Matrix zu beweisen, dass sie der Bedingung NSP_k^γ genügt.

Definition 2.2.3 (RIP-Eigenschaft) Die **restricted isometry** Konstante δ_k der Matrix $M \in \mathbb{R}^{m \times n}$ ist die kleinste Zahl sodass für alle $z \in \Sigma_k$ gilt:

$$(1 - \delta_k) \|z\|_2^2 \leq \|Mz\|_2^2 \leq (1 + \delta_k) \|z\|_2^2.$$

Eine Matrix M erfüllt die **restricted isometry** Eigenschaft der Ordnung k , mit Konstante δ_k , wenn $\delta_k \in (0, 1)$.

Satz 2.2.2 $M \in \mathbb{R}^{m \times n}$ besitze die RIP Eigenschaft der Ordnung k mit $\delta_k \in (0, 1)$, wobei $k = a + b$. Dann besitzt M auch die NSP Eigenschaft der Ordnung a mit

$$\gamma = \sqrt{\frac{a(1 + \delta_k)}{b(1 - \delta_k)}}. \tag{2.11}$$

Beweis: Siehe [10], Lemma 3.3. ■

Der obige Satz sagt somit vor allem aus, dass die RIP-Eigenschaft die NSP-Eigenschaft impliziert. Somit reicht es aus, für eine Matrix die RIP-Eigenschaft zu zeigen, was sich oftmals als leichter erweist.

Zur Abtastung von k -bandlimitierten Signalen bietet sich eine Matrix M also vor allem dann an, wenn sie die RIP-Eigenschaft der Ordnung $a \geq k$ erfüllt.

2.3 Filter auf Graphen

Filter bilden eine allgemeine Klasse von Operationen mit denen auf Signalen gearbeitet werden kann. Typischerweise sollen die Frequenzen eines Signals \mathbf{s} gefiltert werden, indem zum Beispiel diejenigen, die oberhalb einer gegebenen Schranke liegen, abgeschnitten werden. Wie diese Filter auf einem Graphen definiert und genähert werden können möchte ich hier darstellen.

2.3.1 Konstruktion von Filtern

Im Rahmen der Signalverarbeitung auf Graphen werden Filter oft mit Hilfe der Spektralzerlegung der Laplace-Matrix L konstruiert. Dabei wird eine Funktion auf dem Spektrum $\sigma(L)$ von L definiert und mit Hilfe der Fourier-Transformation auf den Graphen ausgeweitet. Dies ist der Inhalt dieses Abschnitts.

Definition 2.3.1 Ein **Graph-Filter** \mathcal{F} ist eine Operation

$$\begin{aligned} \mathcal{F} : \mathcal{S}_G &\longrightarrow \mathcal{S}_G; \\ \mathbf{s} &\longmapsto \mathcal{F}(\mathbf{s}) := \mathbf{s}_{\mathcal{F}} \end{aligned}$$

welche ein gegebenes Signal \mathbf{s} auf ein anderes $\mathbf{s}_{\mathcal{F}}$ abbildet. Dabei ist sowohl \mathbf{s} also auch $\mathbf{s}_{\mathcal{F}}$ auf dem selben Graphen definiert. Der Filter \mathcal{F} ist durch eine Matrix gegeben [15].

Definition 2.3.2 Sei G ein Graph mit Laplace-Matrix L und bezeichne $\sigma(L)$ das Spektrum von L . Eine Abbildung

$$h : \sigma(L) \longrightarrow \mathbb{R},$$

welche die Frequenzen von L in die reellen Zahlen abbildet heißt **Frequenzfilter** auf G .

Definition 2.3.3 Eine Funktion $h_\lambda : \mathbb{R} \rightarrow \{0, 1\}$ heißt **idealer Tiefpassfilter** mit Abschneidefrequenz $\lambda \geq 0$, wenn gilt:

$$h_\lambda(t) = \begin{cases} 1 & \text{wenn } t \in [0, \lambda] \\ 0 & \text{sonst} \end{cases}$$

Die beiden letzten Funktionen genügen der Definition 2.3.1 nicht, da sie nicht auf dem gesamten Graphen operieren. Daher soll als nächstes der Wirkungsbereich dieser Funktionen so erweitert werden, dass sie Signale aus der Knotendomäne von G als Eingabe erhalten können. Sie stellen damit Filter auf G dar.

Das Vorgehen ist dabei wie folgt:

Es sei \mathbf{s} ein auf den Knoten von G definiertes Signal. Dann erlaubt die Darstellung von \mathbf{s} keinerlei Aussagen über die Frequenzen. Durch Übergang zur Spektraldomäne kann die Funktion auf das Spektrum von L angewendet werden und das Resultat mit dem ursprünglichen Signal multipliziert werden. Anschließend kann die Inverse Fourier-Transformation genutzt werden, um erneut ein Signal in der Knotendomäne zu erhalten [7] und [1]. Im Detail heißt das:

- (i) Berechne die Zerlegung $L = U\Lambda U^T$ und wende die Fourier-Transformation an:

$$\hat{\mathbf{s}} = U^T \mathbf{s}$$

Dabei entspricht jetzt $\hat{\mathbf{s}}$ der Darstellung von \mathbf{s} in der Spektraldomäne

- (ii) Durch die Zerlegung von L sind die Eigenwerte Λ verfügbar und der Frequenzfilter kann auf jeden Eigenwert angewendet werden. Das Resultat wird dann mit $\hat{\mathbf{s}}$ multipliziert, um eine Gewichtung durch den Filter zu erhalten.

$$h(\Lambda)\hat{\mathbf{s}} = h(\Lambda)U^T \mathbf{s}.$$

Dabei gilt $h(\Lambda) := \text{diag}[h(\lambda_1), \dots, h(\lambda_n)]$.

- (iii) Schließlich soll eine Darstellung des gefilterten Signals in der Knotendomäne gefunden werden. Dafür bietet sich die inverse Fourier-Transformation an, was eine Multiplikation mit U bedeutet:

$$\mathbf{s}_h = U h(\Lambda)\hat{\mathbf{s}}.$$

Somit wurde ein Filter auf G konstruiert, indem eine Funktion auf $\sigma(L)$ definiert wurde und dann mit Hilfe der Transformationen GFT und IGFT auf den Graphen ausgeweitet wurden. Die obige Überlegung legt folgende Definition nahe:

Definition 2.3.4 Für ein Signal $\mathbf{s} \in \mathbb{R}^n$ auf G und eine Funktion $h : \mathbb{R} \rightarrow \mathbb{R}$ bezeichnet

$$\mathbf{s}_h := U \begin{bmatrix} h(\lambda_1) & & 0 \\ & \ddots & \\ 0 & & h(\lambda_n) \end{bmatrix} U^\top \mathbf{s}$$

das durch h gefilterte Signal.

Abschließend möchte ich folgendes Korollar festhalten, welches im späteren Verlauf eine gewisse Rolle einnehmen wird.

Korollar Sei $\mathbf{s} \in \mathbb{R}^n$ ein k -bandlimitiertes Signal und sei λ_k der k -te Eigenwert von L . Ferner sei h_{λ_k} ein idealer Tiefpassfilter mit Abschneidefrequenz λ_k . Dann folgt:

$$\mathbf{s}_{h_{\lambda_k}} = U \operatorname{diag} [\lambda_1, \dots, \lambda_k, 0, \dots, 0] U^\top \mathbf{s}. \quad (2.12)$$

Tiefpassfilter bieten sich demnach an um k -bandlimitierte Signale zu filtern [1].

2.3.2 Polynomielle Approximation eines Filters

Gegeben sei ein Frequenzfilter $h : \sigma(L) \rightarrow \mathbb{R}$ auf G . In Definition 2.3.4 wurde gezeigt wie die Operation von h auf den gesamten Graphen ausgedehnt werden kann. Dafür wird die Zerlegung von L benötigt, deren Berechnung für große Werte von n viel Zeit und Rechenleistung in Anspruch nimmt. In diesem Abschnitt möchte ich daher aufzeigen, wie die Wirkung eines Filters durch ein Polynom geschätzt werden kann, ohne Einsatz der Matrix U .

Sei dazu

$$\kappa(t) = \sum_{i=0}^d \alpha_i x^i$$

ein Polynom vom Grad d . und $h : \mathbb{R} \rightarrow \mathbb{R}$ eine Funktion, die durch κ approximiert wird. Gemeint ist damit

$$\kappa(x) \approx h(x)$$

für alle x innerhalb eines entsprechenden Bereichs. Dann kann wieder die Fourier-Transformation benutzt werden, um den Wirkungsbereich von κ auf G auszuweiten. Folglich gilt für ein gegebenes Signal \mathbf{s}

$$\kappa \approx h \implies \mathbf{s}_\kappa \approx \mathbf{s}_h.$$

Somit muss κ nur h approximieren, damit selbiges für die gefilterten Signale gilt. Allerdings ist der Tiefpassfilter (2.3.3) nicht stetig aufgrund der Abschneidefrequenz, während ein Polynom stets stetig ist. Daher auch der Name *idealer Tiefpassfilter* in Definition 2.3.3, denn eine stetige Approximation wird im Bereich der Abschneidefrequenz nicht beliebig genau sein können.

Um unnötig viele Notationen zu vermeiden setzte ich für ein Polynom $\kappa(t) = \sum_{i=0}^d \alpha_i x^i$ und eine beliebige Matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$

$$\kappa(\mathbf{A}) := \sum_{i=0}^d \alpha_i \mathbf{A}^i$$

wobei $A^0 := I$.

Satz 2.3.1 Sei L eine Laplacematrix und $\kappa = \sum_{i=0}^d \alpha_i x^i$ ein Polynom vom Grad d , dann gilt:

(i) $\kappa(L) = \kappa(U\Lambda U^\top) = U\kappa(\Lambda)U^\top$.

(ii) $L^i = U\Lambda^i U^\top$ für $i \in \mathbb{N}$.

Beweis:

(i) Für eine beliebige Potenz $i \in \{0, \dots, n\}$ und beliebiges $\alpha_i \in \mathbb{R}$ gilt:

$$\begin{aligned} \alpha_i (U\Lambda U^\top)^i &= \alpha_i \left(U\Lambda \underbrace{U^\top \cdot U}_{=I_n} \Lambda U^\top \cdot \dots \cdot U\Lambda \underbrace{U^\top \cdot U}_{=I_n} \Lambda U^\top \right) = \alpha_i (U\Lambda^i U^\top) \\ &= U (\alpha_i \Lambda^i) U^\top. \end{aligned} \tag{2.13}$$

Folglich erhält man

$$\kappa(L) = \kappa(U\Lambda U^\top) = \sum_{i=0}^d \alpha_i (U\Lambda U^\top)^i = U \left[\sum_{i=0}^d \alpha_i \Lambda^i \right] U^\top = U \kappa(\Lambda) U^\top$$

da die Summe nicht länger von U und U^\top abhängt.

(ii) Folgt aus 2.13 durch Wahl von $\alpha_i = 1$. ■

Damit folgt für κ als polynomielle Approximation an den Frequenzfilter h mit 2.13 [1]:

$$\begin{aligned} \mathbf{s}_\kappa &= U \kappa(\Lambda) U^\top \mathbf{s} \\ &= \sum_{i=0}^d \alpha_i U \operatorname{diag}[\boldsymbol{\lambda}_1^i, \dots, \boldsymbol{\lambda}_n^i] U^\top \mathbf{s} \\ &= \sum_{i=0}^d \alpha_i L^i \mathbf{s}. \end{aligned}$$

Damit kann das Filtern ohne die Zerlegung von L umgesetzt werden, wenn die zugrundeliegende Funktion κ , als polynomielle Näherung von h , gegeben ist. Ferner reduziert sich das Filtern auf Matrix-Vektor Multiplikationen. Wenn L dann dünnbesetzt ist kann das Filtern algorithmisch gut umgesetzt werden [1].

Definition 2.3.5 Für ein Polynom $\kappa = \sum_{i=0}^d \alpha_i x^i$ ist der **polynomielle Filter** κ vollständig spezifiziert durch die folgenden Angaben.

- (i) Die Koeffizienten α_i
- (ii) Den Grad des Polynoms d .

Seine Gestalt ist dann durch die entstehende Matrix $\kappa = \sum_{i=0}^d \alpha_i L^i$ gegeben [7].

3 Das betrachtete Verfahren

An dieser Stelle soll die Arbeit der Autoren nachvollzogen und dargestellt werden. Zuerst wird ein Verfahren zum randomisierten Abtasten bandlimitierter Signale hergeleitet sowie eine geeignete Approximation. Auch hierbei besteht das Ziel darin, ohne Zerlegung von L auszukommen.

Anschließend wird ein Verfahren zur Rekonstruktion der Signale aus den abgetasteten Versionen vorgestellt. Das Verfahren beruht darauf ein dünnbesetztes lineares Gleichungssystem zu lösen und ist damit algorithmisch gut lösbar.

Im Anschluss werde ich darauf eingehen, wie die Methoden in PYTHON implementiert wurden.

3.1 Das Abtasten der Signale

An dieser Stelle soll das in [1] benutzte Verfahren zur Abtastung der Signale genauer dargestellt werden. Wie schon erwähnt basiert die Abtastung auf einer Wahrscheinlichkeitsverteilung \mathcal{P} , gemäß derer abgetastet werden soll. Dem Signal werden demnach zufällig Einträge entnommen. In Anlehnung an (2.6) soll zur Realisierung der Abtastung eine Matrix M genutzt werden. Das eigentliche Verfahren zur Abtastung möchte ich nun darstellen.

Aus der Indexmenge $I := \{1, \dots, n\}$, welche die Knoten des Graphen durchläuft, sollen m Indizes zufällig entnommen werden, wobei $m < n$ gelte. Sei dazu \mathcal{P} eine Wahrscheinlichkeitsverteilung auf I , welche durch den Vektor

$$\mathbf{p} := (\mathbf{p}_1, \dots, \mathbf{p}_n) \in \mathbb{R}^n \quad (3.1)$$

repräsentiert sei. Dabei gelte $\mathbf{p}_i > 0$ für alle $i \in I$.

Die Menge $\Omega = \{\omega_1, \dots, \omega_m\}$ wird nun gewonnen, indem m Indices aus I gemäß \mathbf{p} gezogen werden (mit zurücklegen). Dabei gilt:

$$\mathbb{P}(\omega_j = i) = \mathbf{p}_i$$

für alle $j \in \{1, \dots, m\}$ und $i \in I$ [1]. Die Menge Ω enthält die Indizes an denen das Signal später abgetastet werden soll. Eine Matrix, die den Vorgang der Abtastung umsetzt kann

in jeder Zeile höchstens einen nicht-null Eintrag besitzen. In diesem Eintrag sollte eine 1 stehen damit bei der Matrixmultiplikation nur der gewünschte Eintrag von \mathbf{s} abgetastet wird. Dies führt zur folgenden Definition.

Definition 3.1.1 Gegeben sei ein Graph mit n Knoten und die Menge $\Omega = \{\omega_1, \dots, \omega_m\}$, welche m zufällig gezogene Knoten-Indizes enthalte. Dann besitzt die **Abtastmatrix** $M = (m)_{ij} \in \mathbb{R}^{m \times n}$ folgende Gestalt.

$$m_{ij} := \begin{cases} 1 & \text{wenn } j = \omega_i \\ 0 & \text{sonst} \end{cases} \quad (3.2)$$

für alle $j \in \{1, \dots, m\}$ und $i \in \{1, \dots, n\}$. Das Abtasten eines Signals \mathbf{s} wird dann durch eine Multiplikation mit M durchgeführt und mit $\mathbf{s}_M := M\mathbf{s}$ bezeichnet.

Korollar Sei $\mathbf{s} \in \mathbb{R}^n$ ein Signal auf einem Graphen und M eine Abtastmatrix. Dann gilt für das abgetastete Signal \mathbf{s}_M

$$(\mathbf{s}_M)_j := (\mathbf{s})_{\omega_j}$$

Im Fall von $m \ll n$ ist M entsprechend dünn besetzt, wodurch das Abtasten algorithmisch gut realisiert werden kann.

Es stellt sich die Frage an wie vielen Stellen ein k -bandlimitiertes Signal auf G mindestens abgetastet werden sollte. Um hierfür ein vernünftiges Maß zu haben wird die *gewichtete Graph Kohärenz* eingeführt, welche eine untere Schranke an die Anzahl der zu messenden Einträge m liefern wird.

Definition 3.1.2 Sei $\mathbf{p} \in \mathbb{R}^n$ ein Wahrscheinlichkeitsvektor wie in (3.1) und G ein Graph. Die *gewichtete Graph Kohärenz* der Ordnung k für das Paar (G, \mathbf{p}) ist definiert als

$$v_{\mathbf{p}}^k := \max_{1 \leq i \leq n} \left\{ \mathbf{p}_i^{-1/2} \|U_k^\top e_i\|_2 \right\}.$$

Die Intuition hinter der letzten Definition ist dabei wie folgt begründet:

Gegeben sei das Signal $e_i \in \mathbb{R}^n$, welches im i -ten Eintrag den Wert 1 und sonst den Wert 0 annimmt. Durch Übergang zur Spektraldarstellung $\hat{e}_i = U_k^\top e_i$, erhält man die ersten k Fourier Koeffizienten des Signals. Dabei beschreibt der Quotient

$$\frac{\|U_k^\top e_i\|_2}{\|U^\top e_i\|_2} = \frac{\|U_k^\top e_i\|_2}{\|e_i\|_2} = \|U_k^\top e_i\|_2$$

wie viel Energie von e_i auf den ersten k Fourier Koeffizienten verteilt ist, im Verhältnis zu Verteilung auf allen.

Durch die Betrachtung von $\|U_k^\top e_i\|$ kann jetzt eine Fallunterscheidung vorgenommen werden [1]:

- (i) Wenn $\|U_k^\top e_i\|_2 = 1$ gilt, so gibt es k -bandlimitierte Signale auf G , welches seine Energie ausschließlich auf dem i -ten Knoten annimmt. In diesem Fall sollte besagter Knoten also mit hoher Wahrscheinlichkeit gewählt werden.
- (ii) Gilt hingegen $\|U_k^\top e_i\|_2 = 0$ gibt es kein bandlimitiertes Signal auf G , dessen Wert auf dem i -ten Knoten nicht null ist. Dieser Knoten kann also getrost nicht gewählt werden.
- (iii) In den anderen Fällen gibt $\|U_k^\top e_i\|_2 = \beta \notin \{0, 1\}$, ein Maß für den Einfluss der ersten k Fourier Koeffizienten.

Damit ist klar, dass sich die Einträge von \mathbf{p} an den Werten von $\|U_k^\top e_i\|_2$, $i = 1, \dots, n$ orientieren sollten. Der i -te Eintrag \mathbf{p}_i sollte große Werte annehmen, wenn $\|U_k^\top e_i\|_2$ dies tut. Entsprechend sollten kleine Werte von $\|U_k^\top e_i\|_2$ auch kleine Werte von \mathbf{p}_i bedeuten.

Satz 3.1.1 *Sei G ein Graph und $\mathbf{p} \in \mathbb{R}^n$ repräsentiere eine Wahrscheinlichkeitsverteilung auf den Knoten von G . Ferner sei $1 \leq k \leq n$. Dann gilt für die gewichtete Graph Kohärenz der Ordnung k :*

$$v_{\mathbf{p}}^k \geq \sqrt{k} \tag{3.3}$$

Beweis: *Da die Spalten von U_k normalisiert sind folgt*

$$k = \|U_k\|_{Frob}^2 = \sum_{i=1}^n \|U_k^\top e_i\|_2^2 = \sum_{i=1}^n \mathbf{p}_i \frac{\|U_k^\top e_i\|_2^2}{\mathbf{p}_i} \leq \max_{1 \leq i \leq n} \left\{ \frac{\|U_k^\top e_i\|_2^2}{\mathbf{p}_i} \right\} \sum_{i=1}^n \mathbf{p}_i = (v_{\mathbf{p}}^k)^2.$$

■

Als nächstes soll mit Hilfe von $v_{\mathbf{p}}^k$ gezeigt werden, dass die Abtastmatrix M die *RIP-Eigenschaft der Ordnung k* erfüllt.

Theorem 3.1.1 (RIP-Eigenschaft der Abtastmatrix) *Sei M eine Abtastmatrix konstruiert gemäß der Verteilung \mathbf{p} . Dann gilt für alle $\delta, \epsilon \in (0, 1)$ mit einer Wahrscheinlichkeit von mindestens $1 - \epsilon$:*

$$(1 - \delta) \|\mathbf{s}\|_2^2 \leq \frac{1}{m} \|MP^{-1/2}\mathbf{s}\|_2^2 \leq (1 + \delta) \|\mathbf{s}\|_2^2$$

für alle $\mathbf{s} \in \text{span}(u_1, \dots, u_k)$ wobei

$$m \geq \frac{3}{\delta^2} (v_{\mathbf{p}}^k)^2 \cdot \log \left(\frac{2k}{\epsilon} \right). \quad (3.4)$$

Beweis: Siehe [1], Theorem 2.2. ■

Aus dem letzten Theorem folgt unmittelbar:

- $\frac{1}{m} MP^{-1/2}$ erfüllt die Rip-Eigenschaft der Ordnung k . Folglich sollte mit ihr die Abtastung der Signale durchgeführt werden.
- Da $(v_{\mathbf{p}}^k)^2 \geq k$ müssen mindestens k Einträge abgetastet werden.

Um die Anzahl der benötigten Messungen zu reduzieren sollte die Verteilung \mathbf{p} so gewählt werden, dass $v_{\mathbf{p}}^k$ möglichst kleine Werte annimmt. Dabei ist es sogar direkt möglich eine Verteilung zu wählen, so dass $v_{\mathbf{p}}^k = \sqrt{k}$ gilt. Diese Verteilung ist dann optimal im dem Sinne, dass sie mit der kleinst möglichen Zahl an Messungen auskommt.

Definition 3.1.3 Für einen Graphen G mit entsprechender Laplace-Matrix $L = U\Lambda U^\top$ sei die Verteilung \mathbf{p}^* durch folgende Einträge gegeben

$$\mathbf{p}_i^* := \frac{\|U_k^\top e_i\|_2^2}{k} \quad i = 1, \dots, n. \quad (3.5)$$

\mathbf{p}^* ist dabei optimal in dem Sinne, dass durch die Wahl von \mathbf{p}^* die Graph-Kohärenz den kleinstmöglichen Wert annimmt. Daher wird \mathbf{p}^* ab jetzt **optimale Verteilung** genannt.

Die Optimalität durch das folgende Theorem sichergestellt.

Theorem 3.1.2 (Optimalität von \mathbf{p}^*) Durch die Wahl von \mathbf{p}^* kann die untere Schranke $v_{\mathbf{p}}^k = \sqrt{k}$ immer erreicht werden.

Beweis: Für \mathbf{p}^* muss zum einen geprüft werden, dass es um ein Wahrscheinlichkeitsverteilung handelt und dass $(v_{\mathbf{p}^*}^k)^2 = k$ gilt.

(i) $\sum_{i=1}^n \mathbf{p}_i^* = k^{-1} \sum_{i=1}^n \|U_k^\top e_i\|_2^2 = k^{-1} \|U_k\|_{Frob}^2 = k^{-1}k = 1$, also ist \mathbf{p}^* eine zulässige Wahl.

(ii) $(v_{\mathbf{p}^*}^k)^2 = \left[\max_{1 \leq i \leq n} \{ \mathbf{p}_i^* \|U_k^\top e_i\|_2 \} \right]^2 = \left[\max_{1 \leq i \leq n} \left\{ \frac{\|U_k^\top e_i\|_2}{\|U_k^\top e_i\|_2} \right\} \sqrt{k} \right]^2 = (\sqrt{k})^2 = k$ wodurch die Optimalität bewiesen ist. ■

Dadurch dass die untere Schranke stets erreicht wird, kann Theorem 3.1.1 entsprechend vereinfacht werden.

Korollar Sei M eine Abtastmatrix konstruiert mit \mathbf{p}^* . Dann gilt für alle $\delta, \epsilon \in (0, 1)$ mit einer Wahrscheinlichkeit von $(1 - \epsilon)$:

$$(1 - \delta) \|\mathbf{s}\|_2^2 \leq \frac{1}{m} \|MP^{-1/2}\mathbf{s}\|_2^2 \leq (1 + \delta) \|\mathbf{s}\|_2^2$$

für alle $\mathbf{s} \in \text{spann}(u_1, \dots, u_k)$ mit

$$m \geq \frac{3}{\delta^2} k \cdot \log\left(\frac{2k}{\epsilon}\right).$$

Nachdem die Optimalität von \mathbf{p}^* bewiesen wurde, soll jetzt ein Verfahren zur Approximation hergeleitet werden.

3.2 Approximation der optimalen Verteilung \mathbf{p}^*

In diesem Abschnitt wird beschrieben, wie die optimale Verteilung \mathbf{p}^* (siehe (3.5)) geschätzt werden kann, ohne die Zerlegung von L zu berechnen. Der Ansatz besteht im wesentlichen aus zwei Schritten: Im ersten Schritt wird erläutert wie Einträge von \mathbf{p}^* geschätzt werden können, indem standard-normalverteilte Signale mit Hilfe von κ gefiltert werden. Dabei muss die Abschneidefrequenz λ_k geschätzt werden. Des Weiteren wird die Anzahl der dafür benötigten Signale abgeschätzt.

Im zweiten Schritt wird ersichtlich, wie besagter Eigenwert geschätzt werden kann, ebenfalls durch Anwendung des Filters κ .

Die Autoren folgen dabei einem Ansatz aus [6], indem die Anzahl der Eigenwerte einer hermiteschen Matrix, durch filtern von zufälligen Signalen, geschätzt wird.

3.2.1 Schätzung der Einträge von p^*

Angenommen λ_k sei bekannt und $\mathbf{s} \in \mathbb{R}^n$ sein ein Signal auf G , wobei die Einträge \mathbf{s}_i standard-normalverteilt seien. Wird \mathbf{s} durch den Tiefpassfilter (siehe Definition 2.3.3) h_{λ_k} gefiltert folgt

$$\mathbf{s}_{h_{\lambda_k}} = U \operatorname{diag}[\lambda_1, \dots, \lambda_k, 0, \dots, 0] U^\top \mathbf{s} = U_k U_k^\top \mathbf{s}.$$

Das liefert für den i -ten Eintrag des gefilterten Signals

$$\left(\mathbf{s}_{h_{\lambda_k}}\right)_i = \mathbf{s}_{h_{\lambda_k}}^\top e_i = \mathbf{s}^\top U_k U_k^\top e_i.$$

Hierin liegt der Kern des Verfahrens, denn durch den i -ten Eintrag des Signals kann der gewünschte Wert $\|U_k^\top e_i\|_2^2$ geschätzt werden.

Theorem 3.2.1 Für den Erwartungswert von $\left(\mathbf{s}_{h_{\lambda_k}}\right)_i^2$ gilt:

$$\mathbb{E} \left[\left(\mathbf{s}_{h_{\lambda_k}}\right)_i^2 \right] = \|U_k^\top e_i\|_2^2.$$

Beweis: Zur besseren Lesbarkeit bezeichne $h := h_{\lambda_k}$ den Tiefpassfilter mit Abschneidefrequenz λ_k .

$$\begin{aligned} \mathbb{E} \left[(s_h)_i^2 \right] &= \mathbb{E} \left[\left(e_i^\top U_k U_k^\top \mathbf{s} \right) \left(\mathbf{s}^\top U_k U_k^\top e_i \right) \right] = e_i^\top U_k U_k^\top \cdot \underbrace{\mathbb{E} [\mathbf{s} \mathbf{s}^\top]}_{= \operatorname{Cov}(\mathbf{s}) = I} \cdot U_k U_k^\top e_i \\ &= e_i^\top U_k \underbrace{U_k^\top U_k}_{= I} U_k^\top e_i = e_i^\top U_k U_k^\top e_i \\ &= \|U_k^\top e_i\|_2^2 \end{aligned}$$

da die Einträge von \mathbf{s} standard-normalverteilt sind. ■

Somit erhält man zum Beispiel eine Näherung der optimalen Verteilung indem J zufällige Signale $\mathbf{s}^1, \dots, \mathbf{s}^J$ mit der selben Verteilung wie \mathbf{s} erzeugt und anschließend die Werte $\left(\mathbf{s}_{h_{\lambda_k}}\right)_i^2$ gemittelt werden für alle $i \in \{1, \dots, n\}$.

Dabei soll statt des idealen Tiefpassfilters h_λ die Näherung κ_λ genutzt werden. Allerdings entsteht somit auch ein Approximationsfehler, welcher komponentenweise betrachtet werden kann.

Sei dazu h_λ der ideale Tiefpassfilter von L mit Abschneidefrequenz λ Definiere den Approximationsfehler von κ im i -ten Eintrag durch

$$\hat{e}_\lambda(\lambda_i) := \kappa_\lambda(\lambda_i) - h_\lambda(\lambda_i)$$

und setzte ferner

$$\mathbf{E}_\lambda := \text{diag}[\hat{e}_\lambda(\lambda_1), \dots, \hat{e}_\lambda(\lambda_n)] \in \mathbb{R}^{n \times n}.$$

Dann beschreibt \mathbf{E}_λ den gesamten Approximationsfehler von κ_λ an h_λ .

Theorem 3.2.2 *Seien $\mathbf{s}^1, \dots, \mathbf{s}^J \in \mathbb{R}^n$ J unabhängige, gauss-verteilte Zufallsvektoren mit Erwartungswert 0 und Kovarianzmatrix $J^{-1}\mathbf{I}$ und sei κ_λ eine polynomielle Näherung an den idealen Tiefpassfilter h_λ . Des Weiteren sei j^* der größte Index sodass $\lambda_{j^*} \leq \lambda$ gilt.*

Dann existiert eine Konstante $C > 0$ so dass für alle $\epsilon, \delta \in (0, 1)$ mit einer Wahrscheinlichkeit von mindestens $(1 - \epsilon)$ für die gefilterten Signale gilt:

$$(1 - \delta) \left| \left\| U_{j^*}^\top e_i \right\|_2 - \left\| \mathbf{E}_\lambda U^\top e_i \right\|_2 \right|^2 \leq \sum_{j=0}^J \left(\mathbf{s}_{\kappa_\lambda}^j \right)_i^2 \leq (1 + \delta) \left| \left\| U_{j^*}^\top e_i \right\|_2 + \left\| \mathbf{E}_\lambda U^\top e_i \right\|_2 \right|^2$$

für alle $i \in \{1, \dots, n\}$ unter der Annahme dass gilt

$$J \geq \frac{C}{\delta^2} \cdot \log \left(\frac{2n}{\epsilon} \right).$$

Beweis: Siehe [1], Theorem 4.1. ■

Durch die Wahl von $\kappa_\lambda = h_\lambda$ in Theorem 3.2.2 folgt $\mathbf{E}_\lambda = 0$ und folglich

$$(1 - \delta) \left\| U_k^\top e_i \right\|_2^2 \leq \sum_{j=0}^J \left(\mathbf{s}_{\kappa_\lambda}^j \right)_i^2 \leq (1 + \delta) \left\| U_k^\top e_i \right\|_2^2.$$

Wenn also κ_λ genügend genau h_λ approximiert folgt im Kontext von Theorem 3.2.2:

$$\sum_{j=0}^J \left(\mathbf{s}_{\kappa_\lambda}^j \right)_i^2 \approx \left\| U_k^\top e_i \right\|_2^2, \quad \text{für } i \in \{1, \dots, n\},$$

wodurch eine Näherung für den gewünschten Wert gegeben ist. Dabei beträgt der Fehler in jeder Koordinate mindestens δ .

Schließlich soll nun mit der erhaltenen Näherung die Einträge der optimalen Verteilung geschätzt werden.

Gemäß (3.5) sind die Einträge der optimalen Verteilung \mathbf{p}^* gegeben durch

$$\mathbf{p}_i^* := \frac{\left\| U_k^\top e_i \right\|_2^2}{k} = \frac{\left\| U_k^\top \delta_i \right\|_2^2}{\sum_{i=1}^n \left\| U_k^\top e_i \right\|_2^2},$$

wodurch nun eine Schätzung der Einträge von \mathbf{p}^* gegeben ist durch

$$\tilde{\mathbf{p}}_i := \frac{\sum_{j=0}^J (\mathbf{s}_{c_\lambda}^j)_i^2}{\sum_{i=1}^n \sum_{j=0}^J (\mathbf{s}_{c_\lambda}^j)_i^2}.$$

Damit ist die Verteilung $\tilde{\mathbf{p}}$ bestimmt.

3.2.2 Schätzung des Eigenwertes λ_k

Im zweiten Schritt soll der Eigenwert λ_k geschätzt werden. Dabei wird je eine obere und eine untere Schranke an λ_k angegeben und das so entstehende Intervall zur Approximation genutzt.

Sei dazu λ_k bekannt und h_λ sei der ideale Tiefpassfilter mit Frequenz λ_k . Theorem (3.2.2) besagt in diesem Fall

$$(1 - \delta) \|U_{j^*}^\top\|_2^2 \leq \sum_{j=1}^J (\mathbf{s}_{h_\lambda})_i^2 \leq (1 + \delta) \|U_{j^*}^\top\|_2^2,$$

denn in diesem Fall ist die Approximation exakt, also $E_\lambda = 0 \in \mathbb{R}^{n \times n}$. Ferner folgt durch Summierung

$$(1 - \delta) \sum_{i=1}^n \|U_{j^*}^\top\|_2^2 \leq \sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{h_\lambda})_i^2 \leq (1 + \delta) \sum_{i=1}^n \|U_{j^*}^\top\|_2^2.$$

Wie schon vorher bezeichnet auch hier j^* den größten Eintrag mit $\lambda_{j^*} \leq \lambda_k$. Ferner gilt, da die Spalten von U normalisiert sind

$$\sum_{i=1}^n \|U_{j^*}^\top e_i\|_2^2 = \|U_{j^*}\|_{\text{Frob}}^2 = j^*.$$

Durch zusammenfassen der obigen Ergebnisse folgt somit:

$$(1 - \delta) j^* \leq \sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{h_\lambda})_i^2 \leq (1 + \delta) j^*.$$

Das zeigt, dass die quadrierten Einträge des gefilterten Signals den Wert j^* schätzen. Daher gilt

$$\sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{h_\lambda})_i^2 \approx j^*.$$

Durch Dichotomie kann jetzt ein Intervall der Form $(\lambda_{min}, \lambda_{max})$ gefunden werden, so dass $(k - 1)$ Eigenwerte unterhalb von λ_{min} liegen und k Stück unterhalb von λ_{max} . Demnach liegt der gesuchte Eigenwert λ_k in dem Intervall $(\lambda_{min}, \lambda_{max})$ und kann zum Beispiel durch den Wert

$$\lambda_k \approx \frac{\lambda_{max} + \lambda_{min}}{2}$$

geschätzt werden [1].

Bemerkung: *An dieser Stelle fällt nicht nur der Approximationsfehler von λ_k ins Gewicht, sondern es entstehen weitere Fehler durch die Wahl eines Polynoms κ , welches den Tiefpassfilter approximieren soll.*

Damit sind alle Teile zusammengetragen, um den Algorithmus zu Approximation der optimalen Verteilung zu erstellen:

Im Kern besteht Algorithmus 1 aus zwei Teilen:

1. In Zeile 4 – 10 wird der Eigenwert λ_k geschätzt. Dazu wird λ_k zunächst durch $(\lambda_{max} + \lambda_{min})/2$ geschätzt und als Abschneidefrequenz für κ benutzt. Der Filter wird im Anschluss durch die IF-Abfrage geprüft:
 - a) Sollte die Summe der gefilterten Signale größer als k sein, werden zu viele Einträge beibehalten. Die Abschneidefrequenz ist also zu hoch und wird entsprechend reduziert.
 - b) Im entsprechend anderen Fall werden zu wenige Signale beibehalten und die Abschneidefrequenz wird erhöht.

Anschließend wird die Abbruchbedingung der WHILE-Schleife geprüft. Die Schleife bricht ab, wenn die Abschneidefrequenz genau genug approximiert wurde.

2. Im zweiten Schritt werden die gefilterten Signale genutzt, um die Einträge der geschätzten Verteilung zu Berechnen. Diese werden dann ausgegeben.

Algorithmus 1 : Schätzung der optimalen Abtastverteilung

Input : $G = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ ein Graph mit $|\mathcal{V}| = n$

Bandlimit k

Toleranzniveau $eps \in (0, 1)$

λ_n , Näherung an den größten Eigenwert von L

Output : $\tilde{\mathbf{p}}$, die Schätzung der optimalen Verteilung \mathbf{p}^* .

```

1 Setze  $J = 2 \log(n)$  und erzeuge  $J$  Zufallsvektoren  $\mathbf{s}^1, \dots, \mathbf{s}^J$  wie in Theorem 3.2.2.
2  $\lambda_{max} = \lambda_n$ ,  $\lambda_{min} = 0$ ,  $\lambda = \lambda_n/2$ 
3  $\tilde{\mathbf{p}} = \mathbf{0} \in \mathbb{R}^n$ 
4 while  $round\left(\sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{\kappa_\lambda}^j)_i^2\right) \neq k$  or  $|\lambda_{max} - \lambda_{min}| > eps \lambda_{max}$  do
5     if  $round\left(\sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{\kappa_\lambda}^j)_i^2\right) \geq k$  then
6          $\lambda_{max} = \lambda$ 
7     else
8          $\lambda_{min} = \lambda$ 
9          $\lambda = (\lambda_{max} + \lambda_{min})/2$ 
10    Berechne  $\alpha_i$ , die Koeffizienten von  $\kappa_\lambda$  zur Approximation des Tiefpassfilters  $h_\lambda$ .
11 for  $i = 1, \dots, n$  do
12     $\tilde{\mathbf{p}}_i = \left(\sum_{j=1}^J (\mathbf{s}_{\kappa_\lambda}^j)_i^2\right) / \left(\sum_{i=1}^n \sum_{j=1}^J (\mathbf{s}_{\kappa_\lambda}^j)_i^2\right)$ 

```

3.3 Rekonstruktion k -bandlimitierter Signale

Beim Rekonstruieren eines Signals soll aus einem abgetasteten Signal \mathbf{s}_M das ursprüngliche Signal \mathbf{s} geschätzt werden. Ziel dieses Abschnittes ist es jetzt ein Rekonstruktionsverfahren für k -bandlimitierte Signale herzuleiten.

Dabei ist anzunehmen, dass die erhaltenen Indizes ω_i nicht den exakten Einträgen des Signals entsprechen sondern mit Fehlern behaftet sind, dem sogenannten Rauschen. Der Grund für diese verrauschten Einträge können zum Beispiel Messfehler sein. Der Signalwert wird also durch die Messung leicht verfälscht. Sei dazu $\mathbf{s}_M = M\mathbf{s}$ ein abgetastetes Signal. Dann hat \mathbf{s}_M durch die Messung folgende Gestalt:

$$\mathbf{s}_M = M\mathbf{s} + \mathbf{n}, \quad \mathbf{n} \in \mathbb{R}^m.$$

Dabei wird das Rauschen in der i -ten Koordinate von \mathbf{s}_M durch den i -ten Wert von \mathbf{n}

modelliert, wobei n ein beliebiger Vektor sein kann, die Autoren schränken das Rauschen nicht durch etwaige Annahmen ein [1].

Angenommen eine Basis von $\text{span}(U_k)$ wäre bekannt, dann würde das Lösen von

$$\min_{z \in \text{span}(U_k)} \|Mz - \mathbf{s}_M\|_2 \quad (3.6)$$

eine Rekonstruktion an das ursprüngliche Signal \mathbf{s} liefern. Dies ist ein intuitive Ansatz. Allerdings wurde in (3.1.1) nur für die Matrix $m^{-1} M \mathbf{P}^{-1/2} = m^{-1} \mathbf{P}_\Omega^{-1/2} M$ die RIP-Eigenschaft gezeigt, nicht für M alleine. Dabei ist die Diagonalmatrix $\mathbf{P}_\Omega \in \mathbb{R}^{m \times m}$ wie folgt definiert [1]:

$$(\mathbf{P}_\Omega)_{ii} = \mathbf{p}_{\omega_i}$$

Daher wir in (3.6) eine Gewichtung mit $\mathbf{P}_\Omega^{-1/2}$ vorgenommen, um die RIP-Eigenschaft zu erhalten.

$$\min_{z \in \text{span}(U_k)} \left\| \mathbf{P}_\Omega^{-1/2} (Mz - \mathbf{s}_M) \right\|_2 \quad (3.7)$$

Der nächste Satz zeigt, dass die Lösung von (3.7) eine sinnvolle Rekonstruktion an \mathbf{s} darstellt [1].

Satz 3.3.1 *Sei G ein Graph. und $\Omega = \{\omega_1, \dots, \omega_m\}$ eine Menge von m Indices aus $\{1, \dots, n\}$ unter der Benutzung von $\mathbf{p} \in \mathbb{R}^n$. Dabei sei $m \geq \frac{3}{\delta^2} (v_{\mathbf{p}}^k)^2 \log(\frac{2k}{\epsilon})$. Ferner sei M die zu Ω gehörige Abtastmatrix. und $\epsilon, \delta \in (0, 1)$.*

Dann gilt mit einer von Wahrscheinlichkeit von mindestens $(1 - \epsilon)$ für alle $x \in \text{span}(U_k)$ und für alle $n \in \mathbb{R}^m$:

(i) *Sei x^* die Lösung von (3.7), wobei $\mathbf{s}_M = M\mathbf{s} + n$ gelte. Dann folgt:*

$$\|x^* - \mathbf{s}\|_2 \leq \frac{2}{\sqrt{m(1 - \delta)}} \left\| \mathbf{P}_\Omega^{-1/2} n \right\|_2. \quad (3.8)$$

(ii) *Es existieren Vektoren $n_0 \in \mathbb{R}^m$ sodass für die Lösung von Problem (3.7) mit $\mathbf{s}_M = M\mathbf{s} + n_0$ gilt:*

$$\|x^* - \mathbf{s}\|_2 \geq \frac{1}{\sqrt{m(1 + \delta)}} \left\| \mathbf{P}_\Omega^{-1/2} n_0 \right\|_2. \quad (3.9)$$

Beweis: *Siehe [1], Theorem 3.1.* ■

Wenn keinerlei Rauschen vorliegt (also $n = 0 \in \mathbb{R}^m$) gilt somit

$$\|x^* - \mathbf{s}\|_2 \leq 0 \iff x^* = \mathbf{s}. \quad (3.10)$$

Somit zeigt sich erneut, dass für bandlimitierte Signale eine perfekte Rekonstruktion möglich ist. Gleichsam ist damit gezeigt, dass 3.7 ein gewinnbringender Ansatz zur Rekonstruktion der Signale liefert.

Wenn das Signal mit Rauschen versehen ist, wächst der Rekonstruktionsfehler linear mit einem Faktor von $\|\mathbf{P}_\Omega^{-1/2}n\|_2$

Ein großes Problem beim finden einer Lösung von (3.7) ist die Nebenbedingung

$$z \in \text{span}(U_k)$$

Ohne weiteres lässt sich nicht entscheiden, ob ein gegebener Vektor in $\text{span}(U_k)$ liegt oder nicht. Daher wäre eine Berechnung der Basis U notwendig, dessen Vermeidung allerdings treibende Kraft der bisherigen Überlegungen war.

Um auch weiterhin dieser Linie treu zu bleiben soll die Nebenbedingung durch Einführung eines Strafterms umgangen werden.

Sei dazu

$$g : \mathbb{R} \longrightarrow \mathbb{R} \quad (3.11)$$

ein nicht-negatives, monoton wachsendes Polynom und $\gamma > 0$ eine Konstante. Formuliere das Minimierungsproblem

$$\min_{z \in \mathbb{R}^n} \left\| \mathbf{P}_\Omega^{-1/2} (Mz - \mathbf{s}_M) \right\|_2^2 + \gamma z^\top g(L)z \quad (3.12)$$

Dabei ist $g(L)$ positiv semi-definit und somit ist (3.12) ein konvexes, quadratisches Optimierungsproblem [1, 18].

Der Idee des neuen Ansatzes besteht jetzt darin als Rekonstruktion potentiell alle Signale auf G zu akzeptieren, also auch nicht k -bandlimitierte. Allerdings werden die Signale durch den Term $\gamma z^\top g(L)z$ unterschiedlich gewichtet, in Abhängigkeit von g . Folgendes Beispiel soll dies verdeutlichen.

Angenommen g sei die Identität auf \mathbb{R} . Dann hat der Term die Gestalt

$$\gamma z^\top Lz = \gamma (z^\top U) \Lambda (U^\top z).$$

Dabei enthält Λ die Frequenzen von L . Wenn jetzt das Signal $U^\top z$ viele Nulleinträge enthält, nimmt der Term $(z^\top U) \Lambda (U^\top z)$ einen kleinen Wert an. Mit anderen Worten werden Signale mit hohen Frequenzen stärker bestraft als Signale mit niedrigeren Frequenzwerten, also zum Beispiel bandlimitierte Signale [1].

Da (3.12) ein konvexes und quadratisches Optimierungsproblem ist, sind die Lösungen

durch die Nullstellen der ersten Ableitung gegeben. Eine Lösung von (3.12) kann folglich durch Lösen des Linearen Gleichungssystems

$$\left[M^\top \mathbf{P}_\Omega^{-1} M + \gamma g(L) \right] z = M^\top \mathbf{P}_\Omega^{-1} \mathbf{s}_M \quad (3.13)$$

gefunden werden [1].

Letztlich soll eine Fehlerschranke für die Nutzung der neuen Methode gegeben werden.

Satz 3.3.2 *Sei M eine Abtastmatrix wie in (3.2) und sei $M_{max} > 0$ eine Schranke sodass $\|M\mathbf{P}^{-1/2}\|_2 \leq M_{max}$. Sei $\delta, \epsilon \in (0, 1)$ und sei ferner $m \geq \frac{3}{\delta^2} (v_p^k)^2 \log\left(\frac{2k}{\epsilon}\right)$. Mit einer Wahrscheinlichkeit von mindestens $(1 - \epsilon)$ gilt für alle $\mathbf{s} \in \text{span}(U_K)$, $\gamma > 0$, $n \in \mathbb{R}^n$ und alle nicht-negativen, nicht-fallende Polynome g mit $g(\lambda_{k+1}) > 0$. Sei x^* eine Lösung von (3.7) mit $\mathbf{s}_M = M\mathbf{s} + n$. Dann gilt*

$$\|\alpha^* - \mathbf{s}\|_2 \leq \frac{1}{\sqrt{m(1-\delta)}} \left[\left(2 + \frac{M_{max}}{\sqrt{g(\lambda_k)}} \right) \|\mathbf{P}_\Omega^{-1/2} n\|_2 + \left(M_{max} \sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_k)} \right) \|\mathbf{s}\|_2 \right]$$

und

$$\|\beta^*\|_2 \leq \frac{1}{\sqrt{\gamma g(\lambda_k)}} \|\mathbf{P}_\Omega^{-1/2} n\|_2 + \sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} \|\mathbf{s}\|_2$$

wobei

$$\alpha^* := U_k U_k^\top x^* \quad \text{und} \quad \beta^* := (I - U_k U_k^\top) x^*.$$

Beweis: Siehe [1], Theorem 3.2. ■

Der obige Satz kann genutzt werden, um die gesuchte obere Schranke für den Rekonstruktionsfehler anzugeben.

Korollar (Rekonstruktionsfehler ohne Rauschen) *Sei x^* eine Lösung von (3.7) und $\mathbf{s} \in \text{span}(u_1, \dots, u_k)$ das ursprünglich Signal. Nehme ferner an, dass die Messungen an dem Signal unverfälscht durchgeführt werden. Dann gilt:*

$$\|x^* - \mathbf{s}\|_2 \leq \frac{1}{\sqrt{m(1-\delta)}} \left[M_{max} \sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_k)} \right] \|\mathbf{s}\|_2 + \sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} \|\mathbf{s}\|_2 \quad (3.14)$$

Beweis: Durch Anwendung der Dreiecks-Ungleichung folgt:

$$\begin{aligned} \|x^* - \mathbf{s}\|_2 &= \left\| (U_k U_k^\top x^* - \mathbf{s}) + (I - U_k U_k^\top) x^* \right\|_2 \\ &\leq \|U_k U_k^\top x^* - \mathbf{s}\|_2 + \|(I - U_k U_k^\top) x^*\|_2 = \|\alpha^* - \mathbf{s}\|_2 + \|\beta\|_2. \end{aligned}$$

Der Rest der Aussage folgt durch Einsetzen von $n = 0 \in \mathbb{R}^m$ in die beiden obigen Ausdrücke. ■

Somit ist für den Fall $g(\boldsymbol{\lambda}_k) = 0$ perfekte Rekonstruktion möglich, allerdings folgt dann auch $g(\boldsymbol{\lambda}_1) = \dots = g(\boldsymbol{\lambda}_k) = 0$, da g monoton ist. Wenn $g(\boldsymbol{\lambda}_k) \neq 0$ sollte γ möglichst nahe an 0 gewählt werden und der Ausdruck $\frac{g(\boldsymbol{\lambda}_k)}{g(\boldsymbol{\lambda}_{k+1})}$ sollte minimiert werden, damit der Gesamtfehler der Rekonstruktion minimiert wird.

Sind die Signale durch Rauschen verfälscht und die Funktion g fest, so kann (3.14) optimiert werden, indem γ proportional zu $\|\mathbf{P}_\Omega^{-1/2} n\|_2 / \|\mathbf{s}\|_2$. Um das Resultat noch weiter zu verbessern sollte $g(\boldsymbol{\lambda}_k)$ so klein wie möglich und $g(\boldsymbol{\lambda}_{k+1})$ so groß wie möglich gewählt werden.

3.4 Implementierung des Algorithmus

Ziel dieses Abschnittes ist es die wichtigsten Details der Implementierung darzustellen. Der betrachtete Algorithmus wird von der Wahl des Filterpolynoms κ maßgeblich beeinflusst. Daher soll der verwendete Filter hier dargestellt werden sowie seine Anwendung auf den Signalen. Die verwendeten Bibliotheken sind primär PYGSP [4], eine Bibliothek, die sich mit Signalen auf Graphen in PYTHON beschäftigt und NUMPY [20].

Als Koeffizienten für den Filter wurden Chebyshev- und Jackson-Chebyshev-Koeffizienten genutzt [6]. Dabei sind die Chebyshev-Koeffizienten gegeben durch

$$\alpha_i := \frac{1}{\pi} \frac{\sin [i \arccos(a)] - \sin [i \arccos(b)]}{i}; \quad i > 0.$$

wobei

$$\alpha_0 := \frac{1}{\pi} [\arccos(a) - \arccos(b)]$$

Die Jackson-Chebyshev Koeffizienten vom Grad p sind hingegen wie folgt definiert

$$\alpha_j^p := \frac{\sin(j+1)\alpha_p}{(p+2)\sin(\alpha_p)} + \left(1 - \frac{j+1}{p+2}\right) \cos(j\alpha_p)$$

wobei

$$\alpha_p := \frac{\pi}{p+2}.$$

Das Filterpolynom κ ist somit gegeben durch

$$\kappa = \sum_{i=0}^{30} \alpha_i L^i.$$

Der Grad des Polynoms betrug in allen Anwendungen stets 30, ebenso der Grad der Jackson-Chebyshev Koeffizienten, wenn diese Verwendung fanden.

Für die effiziente Anwendung des Filters ist es wichtig zu erwähnen, dass die Berechnung der Matrix $\kappa = \sum_{i=0}^d \alpha_i L^i$ umgangen werden kann. Nur das Resultat der Multiplikation

$$\mathbf{s}_\kappa = \kappa \mathbf{s} = \left[\sum_{i=0}^d \alpha_i L^i \right] \mathbf{s}$$

ist von Bedeutung. Dadurch kann der Aufwand für das Filtern eines Signals reduziert werden auf $O(d \cdot |\mathcal{E}|)$, wobei $|\mathcal{E}|$ die Anzahl der Kanten in G und d der Grad des Polynoms ist [7]. Zur Realisierung des Filters benutzte ich die Methode `CHEBY_OP` [17] der Bibliothek `PYGSP`, die obige Überlegung umsetzt. Zusammenfassend besteht das Filtern in jedem Schritt aus zwei Schritten

- (i) Im ersten Schritt werden die Chebyshev- oder die Jackson-Chebyshev Koeffizienten berechnet für das Intervall $[0, \lambda]$. Dabei muss das Intervall $[0, \lambda]$ in das Intervall $[-1, 1]$ eingebettet werden. Dies geschieht in den Zeilen 1 – 4.
- (ii) Darauf folgend werden die Koeffizienten α_i und das Signal an die Funktion `CHEBY_OP` übergeben. Der Output ist dann das gefilterte Signal $\mathbf{s}_{\kappa_\lambda}$

Der größte Eigenwert λ_n wurde mit Hilfe der Methode `EIGSH` [21] der Bibliothek `SCIPY` berechnet. Das Maß der Genauigkeit entspricht der default Einstellung.

Durch zusammenfügen der oben genannten Umsetzungen ergibt sich eine Realisierung des Jackson-Chebyshev Filters:

Algorithmus 2 : Jackson-Chebyshev Filter

Input : Signal \mathbf{s}

Geschätzter niedrigster und größter Eigenwert λ_{\min} und λ_{\max}
 λ , die Approximation an den k -ten Eigenwert
 L , die Laplace-Matrix des Graphen.

Output : gefiltertes Signal $\mathbf{s}_{\kappa_\lambda}$

1 $a_1 = (\lambda_{\max} - \lambda_{\min})/2$

2 $a_2 = (\lambda_{\max} + \lambda_{\min})/2$

3 $a = -a_2/a_1$

4 $b = (\lambda - a_2)/a_1$

5 Berechne die Chebyshev / Jackson-Chebyshev Koeffizienten α_k bezüglich a und b .

6 $\hat{\mathbf{s}} = \left[\sum_{k=0}^d \alpha_k L^k \right] \mathbf{s}$.

4 Numerische Experimente

Nachdem die benötigte Theorie und die Implementierung in den vorherigen Kapiteln im Vordergrund standen, soll jetzt das numerische Verhalten genauer untersucht werden. Dabei können die numerischen Experimente in zwei Bereiche eingeteilt werden.

Im ersten Teil bestand mein Vorgehen darin Algorithmus 1 genauer zu betrachten und meine Ergebnisse mit denen der Autoren zu vergleichen. Hierbei bestand mein Ziel zuerst darin die Resultate der Autoren zu reproduzieren, um somit die Richtigkeit meiner Implementierung zu zeigen. Sodann sollte das numerische Verhalten der vorgestellten Methode genauer studiert werden. Zu diesem Zweck sollte der Einfluss der Parameter betrachtet werden. Ferner wollte ich zwei interessante Bemerkungen, die von den Autoren nicht weiter ausgeführt wurden, betrachten. Sie legen eine Möglichkeit nahe wie die Approximation an \mathbf{p}^* und die Rekonstruktion verbessert werden kann. Dazu wurden in der ersten Versuchsreihe nur der Bunny- und der Minnesota-Datensatz verwendet.

Letztlich werden Automobildaten untersucht, um der Frage nachzugehen, inwiefern der Einsatz von Algorithmus 1 auf diesen Daten als sinnvoll bezeichnet werden kann. Hierzu wird der Kontext der Daten umrissen und die entsprechenden Versuche dargestellt.

4.1 Untersuchung der synthetischen Daten

In der ersten Serie von Versuchen wurden die gleichen Daten wie in [1] benutzt. Dabei handelt es sich um den Stanford Bunny, ein 3D Datensatz bestehend aus 2503 Knoten und 130981 Kanten sowie dem Minnesota-Graphen, der dem Straßennetz des US Bundesstaates Minnesota nachempfunden ist. Der Graph besteht aus 2642 zweidimensionalen Knoten und 9257 Kanten. Die Knoten der Graphen sind in der Abbildung 4.1 dargestellt. In [1] wird ferner noch ein weiterer Datensatz untersucht, der Community-Graph [16]. Dieser wurde für die vorliegende Arbeit allerdings nicht berücksichtigt. Der Unterschied in der Wahl bei der des Chebyshev oder des Jackson-Chebyshev Filters wird in dem jeweiligen Abschnitt untersucht.

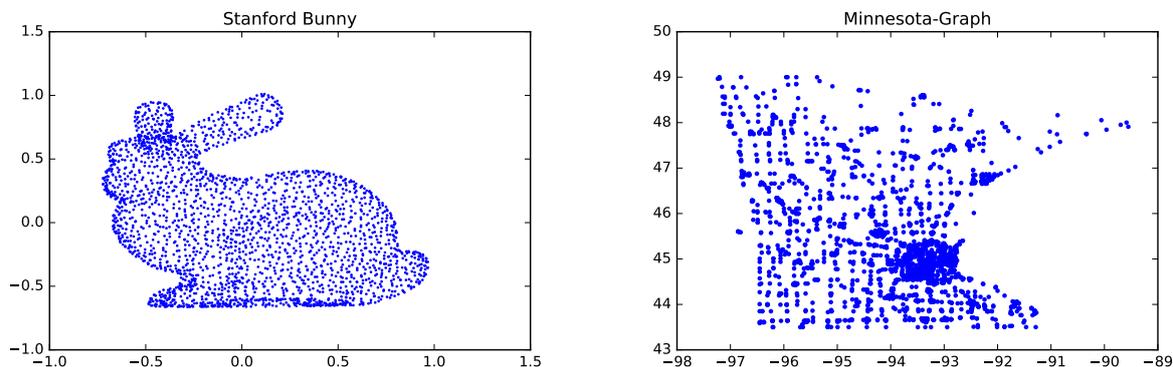


Abbildung 4.1: Die benutzen Datensätze aus der Bibliothek PYGSP.

4.1.1 Der Regularisierungsterm $\gamma z^\top g(L)z$

In Kapitel 3.3 wurde für die Rekonstruktion der Signale folgende Schranke angegeben

$$\|x^* - \mathbf{s}\|_2 \leq \frac{1}{\sqrt{m(1-\delta)}} \left[M_{max} \sqrt{\frac{g(\boldsymbol{\lambda}_k)}{g(\boldsymbol{\lambda}_{k+1})}} + \sqrt{\gamma g(\boldsymbol{\lambda}_k)} \right] \|\mathbf{s}\|_2 + \sqrt{\frac{g(\boldsymbol{\lambda}_k)}{g(\boldsymbol{\lambda}_{k+1})}} \|\mathbf{s}\|_2.$$

Dieser Term soll jetzt genauer betrachtet werden. Zuerst fällt auf, dass die Norm des Signal \mathbf{s} als Faktor auftaucht. Die Normalisierung der betrachteten Signale bietet sich hiernach an, weswegen die Signale ab jetzt als normalisiert seien. Des Weiteren kann die Schranke durch Wahl eines Polynoms der Form

$$g(L) = L^p \quad \text{für } p \in \mathbb{N}$$

verbessert werden [1]. Sei dazu $g(L) = L^p$ mit $p \in \mathbb{N}$ mit $g(\boldsymbol{\lambda}_k) < g(\boldsymbol{\lambda}_{k+1})$. Dann wird mit wachsendem p der Bruch

$$\frac{g(\boldsymbol{\lambda}_k)}{g(\boldsymbol{\lambda}_{k+1})} = \frac{(\boldsymbol{\lambda}_k)^p}{(\boldsymbol{\lambda}_{k+1})^p} = \left(\frac{\boldsymbol{\lambda}_k}{\boldsymbol{\lambda}_{k+1}} \right)^p$$

kleiner. Daher sollte p möglichst groß gewählt werden.

Im Allgemeinen wird durch diese Wahl jedoch gleichzeitig der Wert $g(\boldsymbol{\lambda}_k) = (\boldsymbol{\lambda}_k)^p$ größer, wodurch der zweite Teil des ersten Terms

$$\sqrt{\gamma (\boldsymbol{\lambda}_k)^p}$$

ebenfalls steigt. Mit wachsender Potenz p muss demnach der Wert γ entsprechend klein

gewählt werden. Abbildung 4.2 zeigt das Abfallen des Fehlers bei entsprechender Verkleinerung von γ .

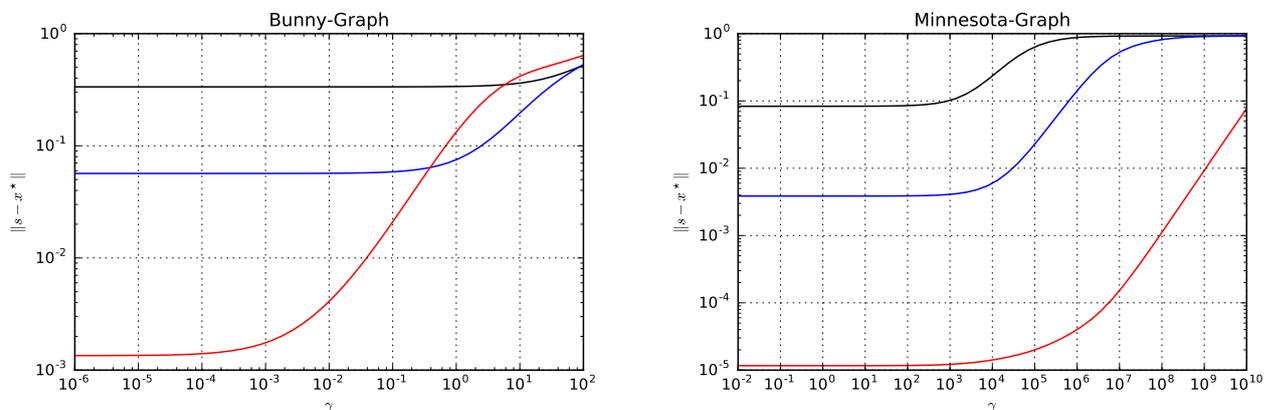


Abbildung 4.2: Fehler der Rekonstruktion in Abhängigkeit von γ . Die schwarze Kurve zeigt den Verlauf bei der Wahl von $g(L) = L^1$, die blaue von $g(L) = L^2$ und die rote von $g(L) = L^4$. Benutzt wurde ein normiertes zufälliges 10-bandlimitiertes Signal sowie der Chebyshev Filter.

Die obigen Überlegungen konnten durch numerische Experimente schnell bestätigt werden. Bei den Versuchen wurden verschiedene Potenzen von L gewählt und der Rekonstruktionsfehler eines zufälligen Signals in Abhängigkeit von γ betrachtet. Auf diese Weise kann eine gute Wahl der Potenz p und des Regularisierungsparameters γ leicht verifiziert werden. Abbildung 4.2 zeigt einen solchen Plot. Die Wahl von $g(L) = L^4$ mit entsprechend kleinem γ lieferte bereits gute Ergebnisse bei den zufälligen Signalen. Der Fehler liegt in diesem Fall etwas zwischen 10^{-5} und 10^{-3} .

Es wurden verschiedene Werte von k getestet, für die jeweils die gleichen Werte von γ und die gleiche Potenz p benutzt wurden. Hierbei zeigten sich ähnlich gute Ergebnisse. Die Wahl des Chebyshev Filters führte in keinem Experiment zu schlechteren Ergebnissen. Schließlich nimmt die Anzahl der Abtastungen m Einfluss auf das Ergebnis Rekonstruktion. Durch Erhöhung von m kann der Fehler minimiert werden, allerdings werden dann immer mehr Punkte des Signals abgetastet, wodurch die Matrix M aus (3.2) immer mehr Einträge hätte, folglich weniger dünn besetzt wäre. Damit würde auch der Aufwand der für das Lösen des Gleichungssystems (3.13) betrieben werden muss höher.

Bemerkung: Für eine feste Wahl von p und γ gilt

$$0 < \lambda_k \implies g(\lambda_k) = (\lambda_k)^p \neq 0$$

Damit ist eine perfekte Rekonstruktion für k -bandlimitierte Signale durch diesem Ansatz nicht mehr möglich.

4.1.2 Die Anzahl

Durch Wahl eines kleinen Toleranzmaßes eps kann die Approximation an den Eigenwert λ_k verbessert werden. Allerdings ist das Maß der Verbesserung abhängig von der Struktur des zugrunde liegenden Graphen. Gleichsam treibt eine Verkleinerung des Toleranzwertes die Anzahl der benötigten Iterationen und somit auch die Laufzeit des Algorithmus stark in die Höhe.

Allerdings kann ein weniger guter Wert von eps durch mehrmaliges Anwenden des Algorithmus kompensiert werden. Von zentraler Bedeutung für die Güte der Approximation ist damit auch die Anzahl der Durchgänge. Ein Durchgang bezeichnet dabei die erfolgreiche Erzeugung der Ergebnisse des Algorithmus. Bei mehreren Durchgängen werden die Einträge der Verteilung komponentenweise addiert und anschließend durch die Anzahl der Durchläufe geteilt. Abbildung 4.8 zeigt dieses Phänomen. Es befindet sich am Ende des Kapitels.

In den numerischen Versuchen konnten zweierlei Dinge schnell falsifiziert werden.

1. Für einen beliebigen aber fest gewählten Toleranzwert eps kann die Approximation, die $\tilde{\mathbf{p}}$ an \mathbf{p}^* leistet, durch Erhöhung der Durchläufe verbessert werden.

Dabei bezeichnet \mathbf{p}^* die optimale Verteilung auf dem Graphen und $\tilde{\mathbf{p}}$ die Geschätzte.

2. Besagte Verbesserung wird durch Erhöhung mehrmaliges Ausführen nicht beliebig gut, sondern strebt tendenziell einer unteren Schranke entgegen.

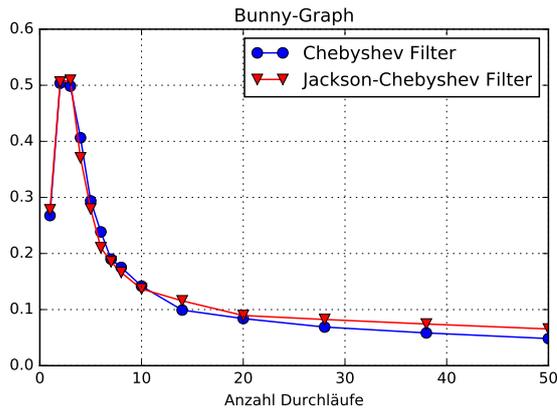
Für die Fehlermessung der Verteilungen wurde der absolute und der relative Fehler benutzt. Der Fehler wurde in jeder Koordinate gemessen, die Ergebnisse summiert und Anschließend gewichtet mit der Anzahl der Knoten n . Formal ergibt sich der absolute Fehler

$$(\mathbf{p}^*, \tilde{\mathbf{p}}) \mapsto \frac{1}{n} \sum_{i=1}^n \|\mathbf{p}_i^* - \tilde{\mathbf{p}}_i\|_1 \quad (4.1)$$

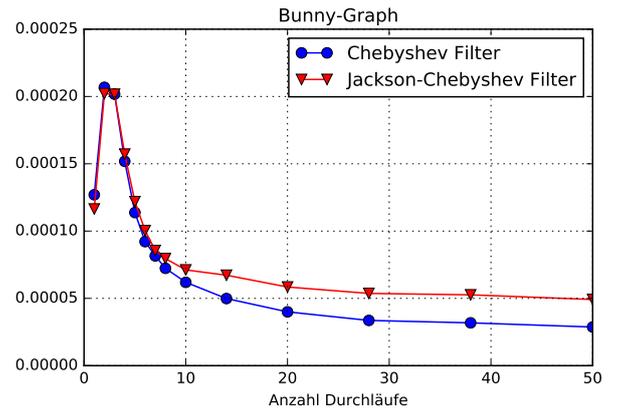
sowie relativer Fehler

$$(\mathbf{p}^*, \tilde{\mathbf{p}}) \mapsto \frac{1}{n} \sum_{i=1}^n \frac{\|\mathbf{p}_i^* - \tilde{\mathbf{p}}_i\|_1}{\|\mathbf{p}_i^*\|_1} \quad (4.2)$$

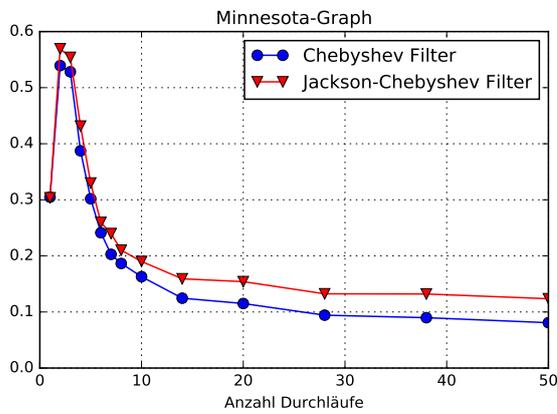
Abbildung 4.3 visualisiert die Fehler für den Bunny- und den Minnesota-Graphen. Die Plots belegen die beiden obigen Behauptungen. Die Approximation wird verbessert, auch wenn nicht beliebig gut. Zu betonen ist das anfängliche Ansteigen des Fehlers bei einer kleinen Anzahl der Durchgänge. Ein Wahl von 10 liefert bereits eine sichtbare Verbesserung. Ebenfalls ersichtlich werden die erneuten schlechteren Ergebnisse bei der Wahl der Jackson-Chebyshev Koeffizienten. Bei der Analyse von anderen Datensätzen der Bibliothek pygsp [17] konnten ähnliche Ergebnisse erzielt werden.



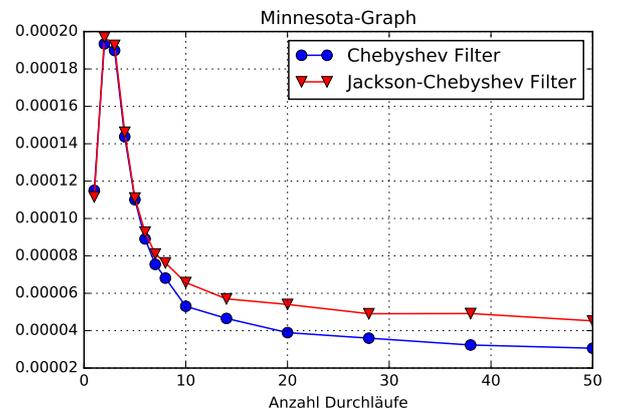
(a) Relativer Fehler



(b) Absoluter Fehler



(c) Relativer Fehler



(d) Absolute Fehler

Abbildung 4.3: Relativer und absoluter Fehler des Bunny-Graphen

4.1.3 Der Einfluss von k und eps

Der Einfluss, der durch den Parameter eps ausgeübt wird, ist schwierig zu fassen. Zum einen zeigte sich während der Betrachtungen der Fehler dass die konkrete Wahl von eps kaum bis gar keinen Einfluss auf den Fehler nahm. Sogar durch eine Wahl von $eps = 1.0$, was im Grunde kaum als Toleranzniveau bezeichnet werden kann, änderte sich die Größenordnung der Fehler kaum. Ähnliches ergab die Untersuchung des Rekonstruktionsfehlers. Der Parameter schien keinen Einfluss auf die Plots zu haben. Im Gegensatz dazu konnte eine Änderung der Visualisierungen betrachtet werden, wenn der Parameter verschiedene Werte annimmt. Hier zeigte sich eine Verbesserung der Approximation. Abbildung 4.4 zeigt diesen Sachverhalt. Genauere Betrachtungen zeigten, dass die Verbesserung nicht an allen Stellen besser werden, sondern lediglich an denen, die von der optimalen Verteilung \mathbf{p}^* mit einer hohen Wahrscheinlichkeit bewertet werden. Die letzte Beobachtung erklärt damit den fehlenden Einfluss bei der Untersuchung des Fehlers: Die Anzahl der Einträge in denen der Fehler kleinere Werte annimmt fallen im Vergleich zu den Stellen, an denen sich dieses Verhalten nicht zeigt zu wenig ins Gewicht.

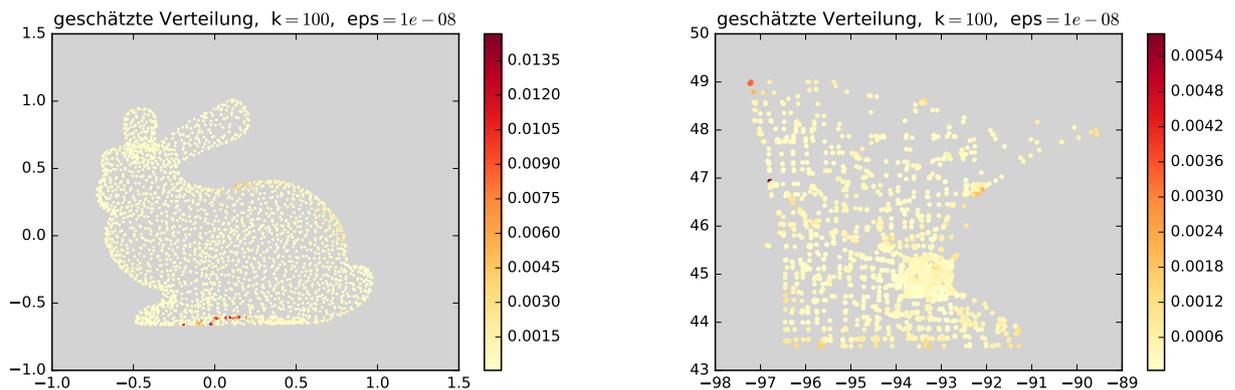


Abbildung 4.4: Geschätzte Verteilung nach einem Durchgang mit $eps = 10^{-8}$. Eine leichte Verbesserung zu den beiden mittleren Bildern der Abbildung 4.8 ist dabei zu erkennen.

Der Wert k fungiert weniger als Parameter, sondern eher als fester Input. Während das Prüfen von verschiedenen Werten von eps noch zweckmäßig sein kann, sollte der Wert für k von Beginn an feststehen.

4.2 Automobildaten im Fraunhofer SCAI

Innerhalb des Entwicklungsprozesses neuer Automobile spielen Sicherheitsaspekte eine große Rolle. Ein klassisches und sehr bekanntes Verfahren ist der Crashtest, bei dem ein Unfall unter Laborbedingungen durchgeführt und anschließend analysiert wird. Auf diese Weise können Annahmen an die Fahrzeugsicherheit bestätigt oder widerlegt werden. Allerdings sind die Kosten für solche Test oft sehr hoch. Eine Möglichkeit diese Kosten zu umgehen stellen numerische Simulationen dar, mit denen verschiedene Aspekte eines Zusammenpralls durch unterschiedliche Modelle simuliert werden. Dabei werden unter anderen partielle Differentialgleichungen zur Modellierung benutzt und anschließend numerisch gelöst. Hierfür bietet sich die Methode der Finiten Elementen an [12].

Mittels dieser Simulationen können verschiedene Parameter wie die Dicke oder Geometrie eines Bauteils studiert werden, ohne einen realen Versuch durchzuführen [8]. Die Ergebnisse der Simulationen sind dabei sehr genau, was wiederum einen häufigen Einsatz der Simulationen innerhalb eines Entwicklungsprozesses mit sich bringt [14].

Das Ziel dieses Prozesses liegt darin einen Prototyp iterativ so anzupassen, dass zum einen funktionale Aspekte erfüllt werden und zum anderen Nebenaspekte wie Sicherheit, ein festes Budget oder gesetzliche Richtlinien eingehalten werden. Die Analyse der Ergebnisse wird typischerweise durch den Einsatz von Visualisierungssoftware vorgenommen [9].

Um die Analyse zu vereinfachen wird der simulierte Crash-Test in mehrere Zeitschritte eingeteilt. Die Simulationsergebnisse bestehen somit aus einer Reihe von Zeitschritten und den Koordinaten der Messpunkte. Jede einzelne Simulation erzeugt somit ($\#$ Zeitschritte \cdot $\#$ Messpunkte) viele Ergebnisdaten [14]. Werden mehrere Simulationen durchgeführt, steigt die Anzahl um den entsprechenden Faktor. Ferner werden die Simulationen, mit wachsender Rechenleistung, immer aufwendiger und die Menge an Ergebnisdaten immer größer. Ein Phänomen, welches im Allgemeinen als *Big Data* bekannt ist.

Aufgrund der schieren Masse an Simulationsdaten stellte die Analyse der Ergebnisse eine völlig neue Herausforderung dar, die nicht mehr von einem einzelnen Ingenieur durchgeführt werden kann [12]. Hierfür werden computergestützte Verfahren zur Verarbeitung der Daten eingesetzt. Dennoch können im Allgemeinen nicht alle Daten analysiert werden, da der hierfür betriebene Aufwand zu hoch ist. Es bietet sich vielmehr an die Menge der zu untersuchenden Daten zu reduzieren, durch den Einsatz von verschiedenen Methoden. Vor der Analyse könnte der Einsatz von Clusteralgorithmen die Daten

in verschiedene Teilmengen einteilen, die dann gesondert analysiert werden [12].

Des Weiteren könnten Methoden der Dimensionsreduktion eingesetzt werden, welche darauf zielen die hochdimensionalen Daten in einen Raum mit einer niedrigeren Dimension einzubetten. Betrachtet man die frei wählbaren Parameter als Datenpunkt in einem Raum, so entsprechen die Simulationen hochdimensionalen Punkten in einem geeigneten Raum, die durch die Reduktion in einen Raum niedrigerer Dimension eingebettet werden können. Die Daten können dann in dem Raum mit der niedrigeren Dimension analysiert werden.

Eine dritte Möglichkeit könnte darin besteht die Daten durch Abtasten zu reduzieren [12]. Wenn die ursprünglichen Daten gut genug rekonstruiert werden können, kann der Speicherbedarf der Messpunkte reduziert werden, indem lediglich die abgetasteten Daten gespeichert werden. In diesen Rahmen fällt der letzte Teil dieser Arbeit.

Verspricht die Anwendung des Verfahrens aus [1] einen Vorteil bei der Anwendung auf Automobildaten?

Hierfür müssten die Daten in der Kontext der Signalverarbeitung auf Graphen eingebettet werden können, also zum Beispiel k -bandlimitiert für ein geeignetes k , und genügend genau rekonstruierbar sein.

Um der oben aufgeworfenen Frage nachzugehen untersuchte ich das Verhalten von Algorithmus 1 auf einem Datensatz aus 131 Modellen zu je 17 Zeitschritten. Dabei bezeichnet ein Model eine feste Parameterwahl [12]. Für jeden Zeitschritt sind jeweils die xyz -Koordinaten der Messpunkte im \mathbb{R}^3 verfügbar, die Struktur des Graphen ist durch das FEM-Gitter gegeben. Damit steht der Anwendung von Algorithmus 1 nichts mehr im Wege.

Das betrachtete Kraftfahrzeug ist ein Chevrolet Pickup C2500, es handelt sich dabei um einen bis 2015 frei verfügbaren Datensatz. Innerhalb eines Models ist jedem Bauteil eine Identifikationsnummer zugeordnet. In dieser Arbeit soll jetzt exemplarisch das Bauteil mit der Identifikationsnummer 2000002 betrachtet werden. Es handelt sich dabei um einen Längsträger. Sollten sich die Ergebnisse als geeignet herausstellen, käme eine Ausweitung der Analyse auf das ganze oder andere Modelle in Betracht. Abbildung 4.5 zeigt den Chevrolet und das verwendete Bauteil.

Während meiner Analysen betrachtete ich zahlreiche Modelle und untersuchte die Ergebnisse in verschiedenen Parametereinstellungen und Zeitschritten. Dabei zeigte sich eine große Redundanz innerhalb der Ergebnisse. In Absprache mit meinen Betreuern werde ich daher nur die Ergebnisse des Bauteils 2000002 aus dem ersten Modell dokumentieren. Ferner entschied ich mich für eine $2D$ -Darstellung der Ergebnisse, auch wenn das ursprüngliche Bauteil ein $3D$ -Datensatz ist.

Die Methode der Untersuchung

Als Signal zu einem festen Zeitschritt wurden jeweils die x -Koordinaten der Knoten interpretiert. Der so entstehende Vektor wurde normiert, abgetastet und anschließend wieder Rekonstruiert. Analog bin ich bei den y - und den z -Koordinaten vorgegangen.

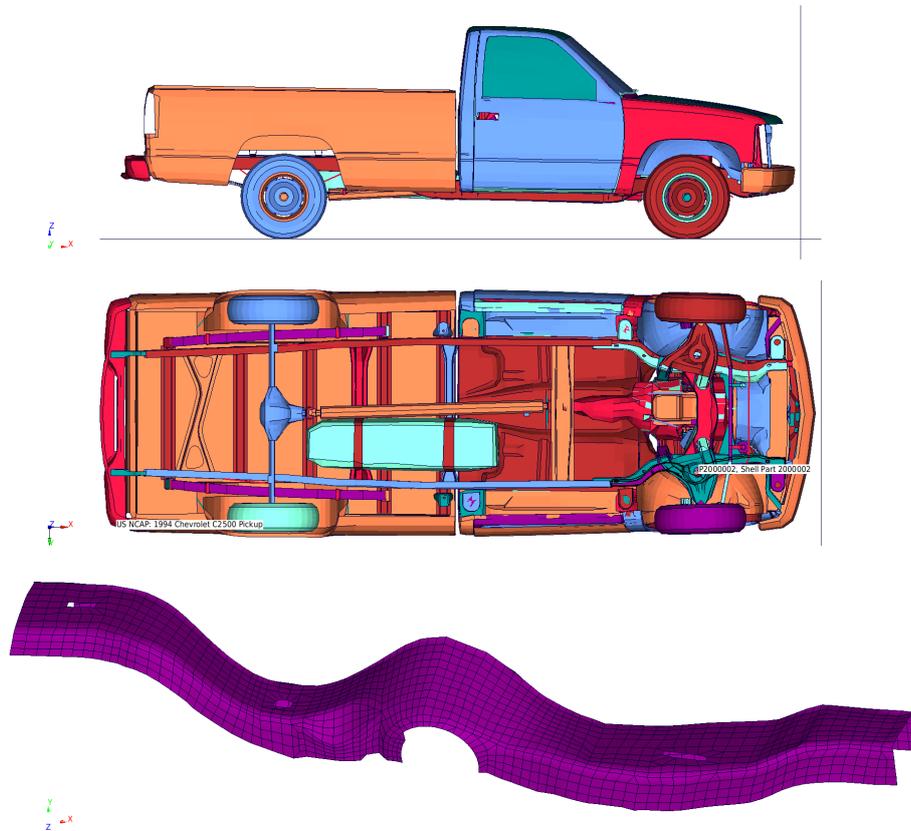


Abbildung 4.5: Chevrolet C2500 Pickup. Im untersten Bild ist das Bauteil 2000002 aus der Nähe dargestellt, im mittleren Bild ist es leicht hervorgehoben.

Das Abtasten der Signale stellt eine Komprimierung der Datenmenge dar, wobei die Einsparung umso höher wäre, je weniger Abtastungen vorgenommen werden. Da bei einem k -bandlimitierten Signal mindestens k Messungen vorgenommen werden müssen, wäre hier ebenfalls ein kleiner Wert von Interesse.

Sollten sich die oben beschriebenen Signale als bandlimitiert auffassen lassen und wäre die Rekonstruktion gut genug so könnte die Anwendung von Algorithmus 1 als erfolgreiche Komprimierung der Knotenmenge gesehen werden.

Die Autoren stellen allerdings keine Methode bereit um für ein gegebenes Signal \mathbf{s} und einen gegebenen Wert k zu prüfen, ob \mathbf{s} ein k -bandlimitiertes Signal darstellt. Ferner konnte ich im Rahmen meiner Recherche keine solche Methode ausfindig machen. Daher entschied ich mich für folgendes Vorgehen:

- (i) Um zu prüfen ob es sich bei den oben beschriebenen Signalen um k -bandlimitierte Signale handelt, sollen verschiedene Werte von k getestet werden, indem die Güte der Approximation betrachtet wird.

Die ersten numerischen Tests lieferten gute bis sehr gute Ergebnisse bei der Re-

konstruktion von zufälligen bandlimitierten Vektoren. Daher sollte ein Signal als k -bandlimitiert gelten, wenn es bei der entsprechenden Rekonstruktion genügend gute Ergebnisse liefert.

- (ii) Neben dem obigen Vorgehen sollte die Güte der Approximation allgemein untersucht werden. Dabei sollte weniger der Wert von k im Vordergrund stehen, als vielmehr die Frage, ob die Ergebnisse der Rekonstruktion die Anwendung der Abtastung rechtfertigen.

4.3 Untersuchung des Bauteils 2000002

Innerhalb meiner Arbeit habe ich zahlreiche Modelle sowie verschiedene Bauteile und Parametereinstellungen untersucht. Wie schon erwähnt zeigte sich dabei eine große Redundanz innerhalb der Ergebnisse.

Der letzte Teil dieses Kapitel stellt jetzt meine Ergebnisse exemplarisch dar. Dabei werde ich stets den gleichen Zeitschritt, den 7., betrachten. Die Ergebnisse auf den übrigen Zeitschritten sind redundant, da nur die Koordinaten betrachtet wurden und nicht das Maß der Verformung bei einem Wechsel der Zeitschritte. Die Analyse wurde in PYTHON durchgeführt. Zum besseren Überblick ist das Bauteil in Abbildung 4.6 in 2D Darstellung zu sehen.

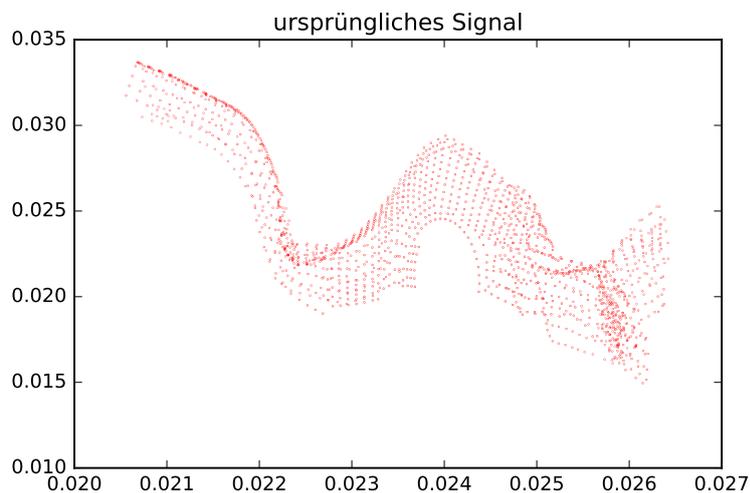


Abbildung 4.6: Die Knotenmenge des Bauteils 2000002 im 7. Zeitschritt. Dabei ist auf der rechten Seite die Verformung zu sehen.

Zuerst ging ich der Frage nach, ob eine Reproduktion der Ergebnisse aus der ersten Testreihe auf dem Bauteil möglich ist. Dabei konnte ich in sämtlichen Test sehr ähnliche Resultate erzielen, vor allem bei der Rekonstruktion von zufälligen bandlimitierten Signalen.

Sodann wurden, durch Betrachtung der Fehlerplots, der Parameter γ und die Potenz p bestimmt. Dabei ergaben sich folgende Werte

- Für die x -Koordinaten ergab sich $\gamma \geq 10^7$.
- Für die y -Koordinaten ergab sich $\gamma \leq 10^{-2}$.
- Für die z -Koordinaten sollte $\gamma \in [10^6, 10^{17}]$ liegen.

Dabei stellte sich je eine Potenz von $p = 1$ als beste Wahl heraus. Der Fehler orientierte sich in diesem Kontext um den Wert 10^{-1} . Damit passen diese Ergebnisse wenig zu denen aus der ersten Testreihe, wo ein möglichst kleiner Wert von γ und eine hohe Potenz empfohlen wird.

Diese Parameter nutzend untersuchte ich die Wiederherstellung des Bauteils und den Einfluss der übrigen Parameter auf die Ergebnisse. Zuerst wählte ich feste Werte von k aus und teste verschiedene Parametereinstellungen in unterschiedlichen Zeitschritten. Dabei konnte ich schnell festhalten, dass durch die Anzahl der Abtastungen enormer Einfluss auf die Wiederherstellung genommen werden kann. Abbildung 4.9 zeigt dieses Verhalten. In dieser Versuchsreihe betrug $eps = 10^{-4}$ und $k = 100$. Es wurde je ein Durchlauf durchgeführt. Dabei zeigte sich allerdings auch, dass eine sehr hohe Anzahl von Messungen nötig ist, um passable Ergebnisse zu erhalten. Dieses Verhalten zeigte sich auch auf den anderen Modellen und auf verschiedenen Zeitschritten.

Bei dem Studium der Durchläufe konnte ich Ähnliches feststellen. Hierbei ist eine leichte Verbesserung der Ergebnisse feststellbar, allerdings ist der dadurch genommene Einfluss deutlich kleiner. Auch hier zeigten sich redundante Ergebnisse. Eine Testreihe mit unterschiedlich hohen Anzahlen ist in Abbildung 4.10 abgebildet.

Wie auch schon in der ersten Testreihe nimmt das Toleranzniveau eps nur wenig bis gar kein Einfluss. Auch hier besteht eine mögliche Interpretation darin, dass eps nur an wenigen Stellen Einfluss auf die Schätzung nimmt und somit den globalen Fehler kaum verändert. Abbildung 4.11 zeigte eine kurze Testreihe, die das Wirken von eps beschreibt.

Als Zwischenfazit konnte ich die Anzahl der Abtastungen und die der Durchläufe als entscheidende Parameter benennen, die einen sichtbaren Einfluss auf die Rekonstruktion nehmen. Hierbei zeigte sich eine Verbesserung der Ergebnisse durch Erhöhung der Werte. Im Rahmen der ersten Versuchsreihe konnte ich keinen optimalen Wert für k bestimmen. Zwar zeigten sich bessere Ergebnisse, bei der Wahl von $k \approx 100$ als bei $k \approx 10$ allerdings konnten die Ergebnisse der Rekonstruktion nicht gänzlich überzeugen.

Ziel der zweiten Reihe von Tests war die Bestimmung eines geeigneten Wertes für k , damit die Signale als k -bandlimitiert betrachtet werden können. Hierzu wurde erneut

die Rekonstruktion betrachtet, allerdings diesmal mit einer Durchgangszahl von 50 und einer erhöhten Zahl der Messungen m . Dabei wurde für jeden Wert von k drei Versuche durchgeführt. Bei der ersten wurden 500 Abtastungen vorgenommen, bei der zweiten 1000 und bei der dritten 1500. Abbildung 4.7 zeigt eine solche Testreihe für die Wahl von $k = 100$. Auch hier konnte keine befriedigender Wert für k gefunden werden.

Unabhängig von der Suche nach einem guten Wert für k stellte sich somit heraus, dass die geleistete Güte der Approximation als nicht ausreichend bezeichnet werden muss. Durch entsprechende Wahlen der Parameter kann sie zwar deutlich verbessert werden, allerdings steht der damit verbundene Mehraufwand in keinem Verhältnis. So erscheint die Rekonstruktion nach 1500 Abtastungen zwar gut zu sein. Allerdings werden dann lediglich 214 nicht abgetastet, was den Einsatz zur Komprimierung fraglich erscheinen lässt.

An dieser Stelle möchte ich auch erwähnen, dass die Rekonstruktion selbst mit der optimalen Verteilung und 1714 Messungen keine befriedigenden Ergebnisse erzielen konnte. Abschließend kann ich festhalten, dass ich in keiner Testreihe und durch keine Parameterwahl Ergebnisse erzielen konnte, die den Ansprüchen gänzlich genügten trotz Verbesserung der Ergebnisse oder Nutzung der optimalen Verteilung \mathbf{p}^* . Somit kam ich zu dem Schluss, dass der Einsatz von Algorithmus 1 keine gewinnbringenden Ergebnisse liefert. Für die Regularisierung kann ich festhalten, dass die sehr hohen Werte von γ und die kleine Potenz $p = 1$ im Widerspruch zu den vorherigen Ergebnissen stehen. Diese Ergebnisse sind mit keinem vorherigen Versuch vergleichbar, sie widersprechen ihnen eher. Die Überlegungen den Regularisierungsterm betreffend gelten allerdings nur, wenn das Signal bandlimitiert ist.

Da ferner kein geeigneter Wert für k durch die anderen Versuche ausfindig gemacht werden konnte, kam ich hier zu dem Schluss, dass es sich bei den xyz -Koordinaten nicht um bandlimitierte Signale handelt.

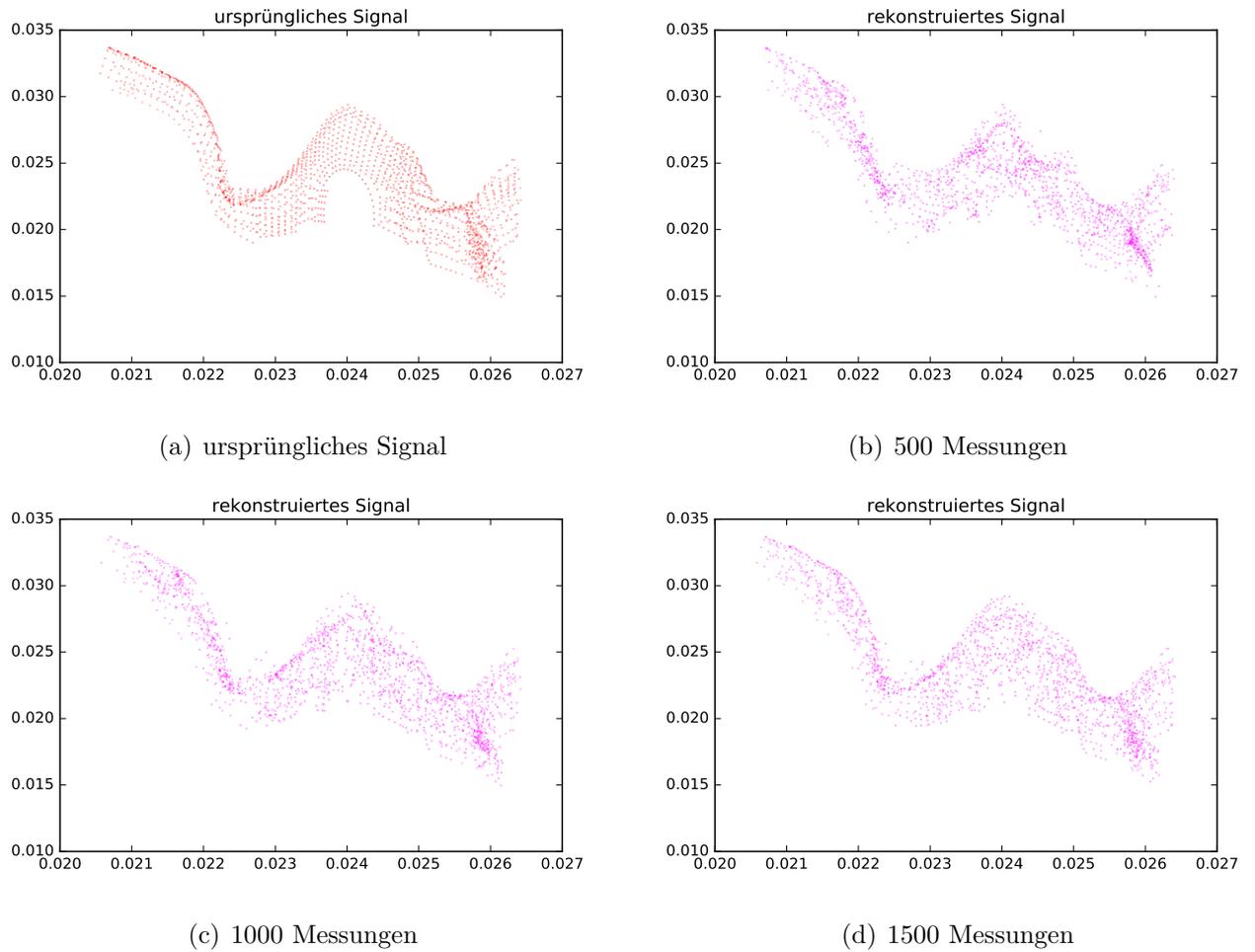
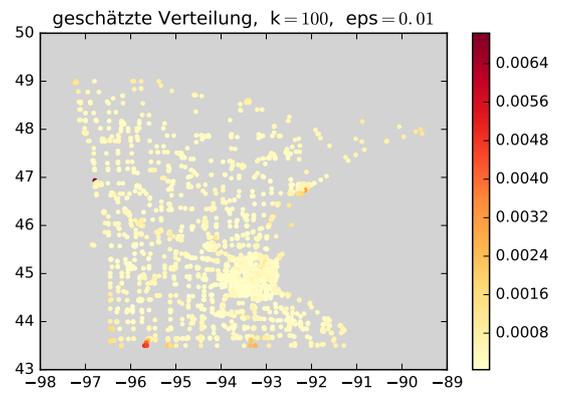
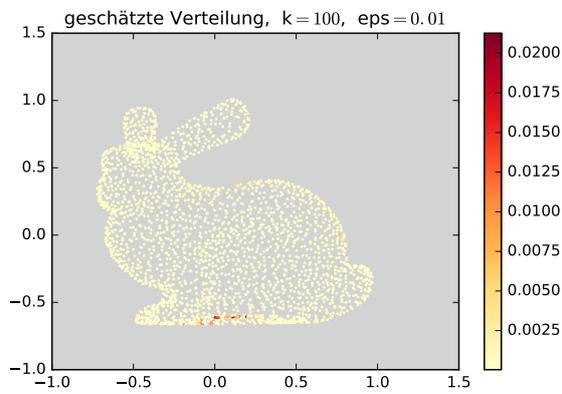
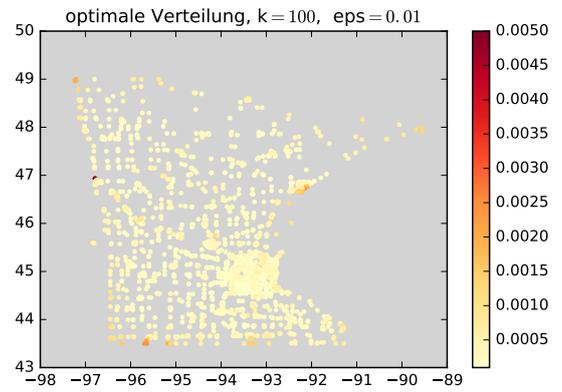
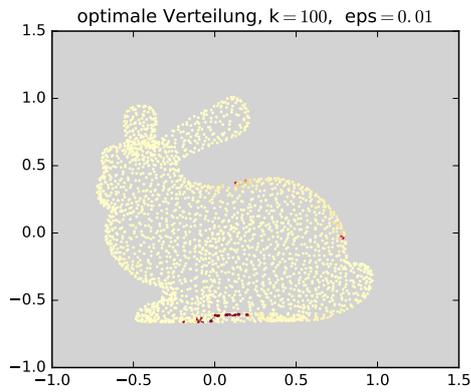
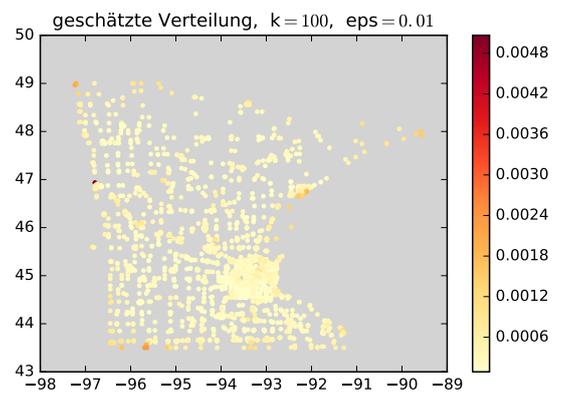
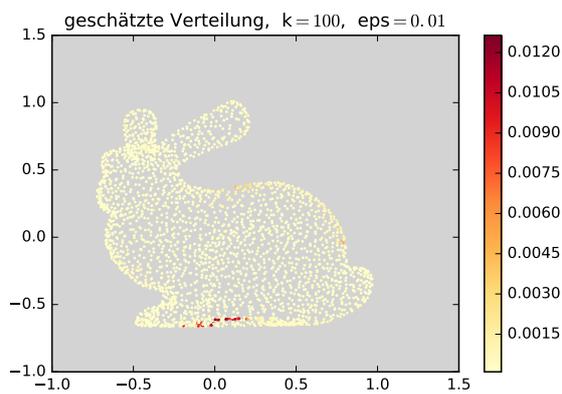


Abbildung 4.7: Verschieden viele Abtastungen bei festem Wert von $k = 100$. Dabei wurden je 50 Durchgänge bei einem Toleranzniveau von $eps = 10^{-8}$.



(c) Approximation nach dem ersten Durchgang

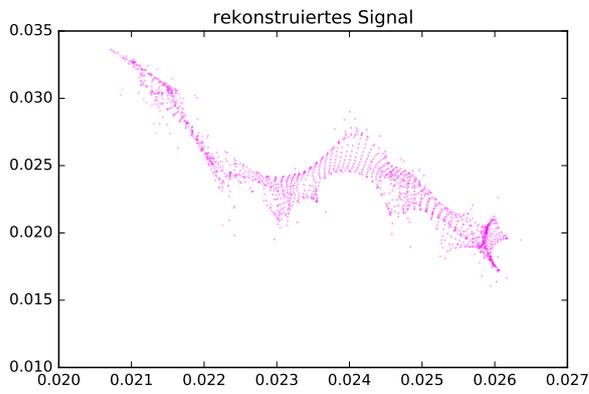
(d) Approximation nach dem ersten Durchgang



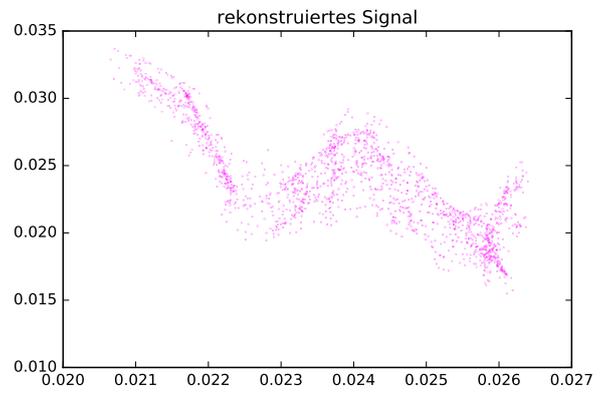
(e) Approximation nach 10 Durchgängen

(f) Approximation nach 10 Durchgängen

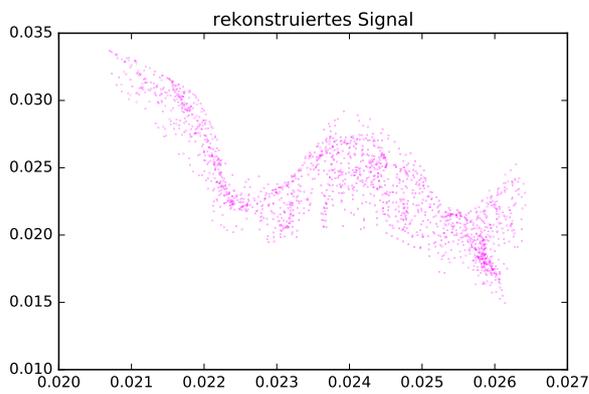
Abbildung 4.8: Visualisierung der Wahrscheinlichkeitsverteilungen. Die Linke Spalte zeigt die Verteilung auf dem Bunny-Graphen, die rechte die des Minnesota-Graphen. Rote Punkte werden mit einer hohen Wahrscheinlichkeit gewählt, gelbe Punkte mit entsprechend niedrigerer.



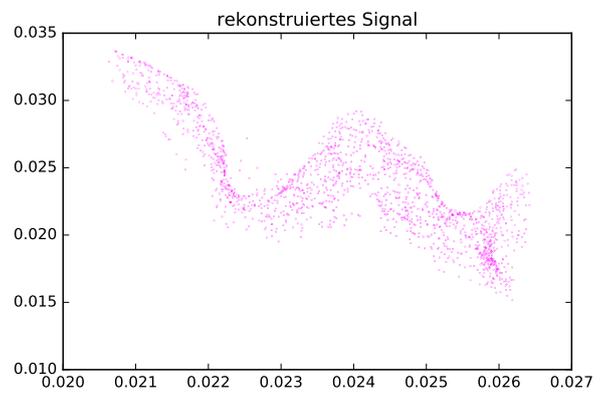
(a) 100 Messungen



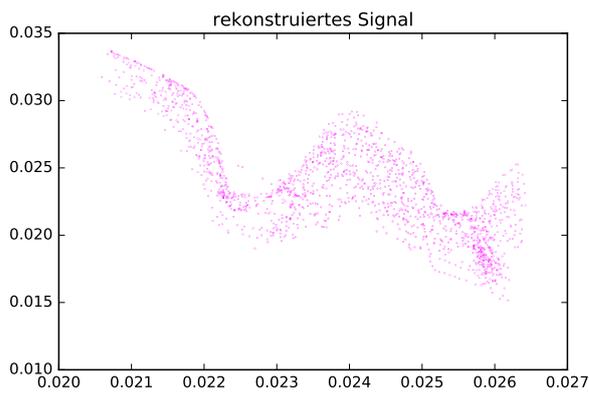
(b) 500 Messungen



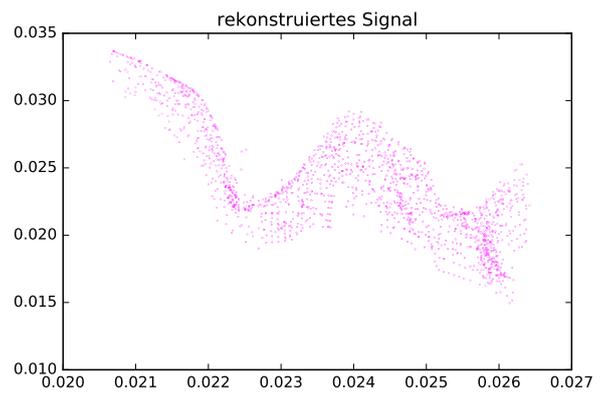
(c) 800 Messungen



(d) 1000 Messungen

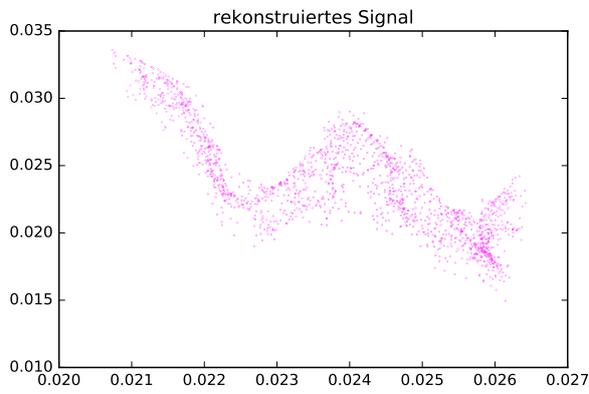


(e) 1500 Messungen

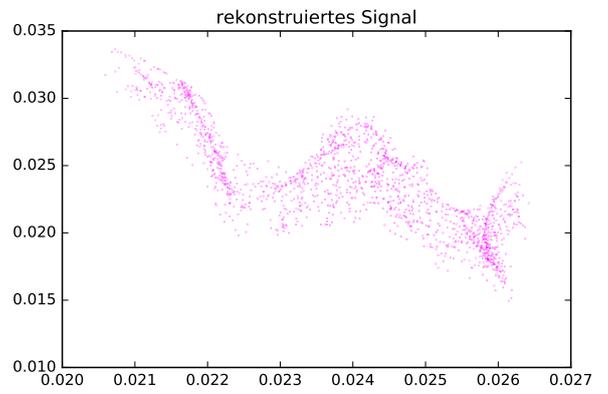


(f) 1714 Messungen

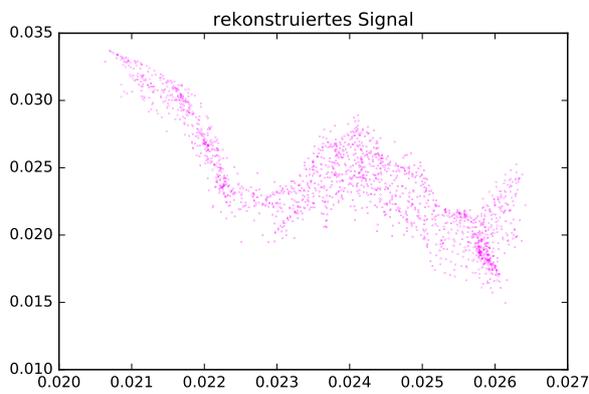
Abbildung 4.9: Einfluss der Anzahl der Messungen auf das Ergebnis der Rekonstruktion. Dabei ist eine deutliche Verbesserung feststellbar.



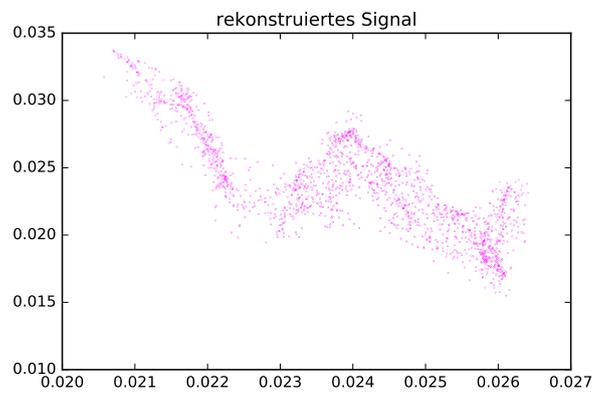
(a) 1 Durchlauf



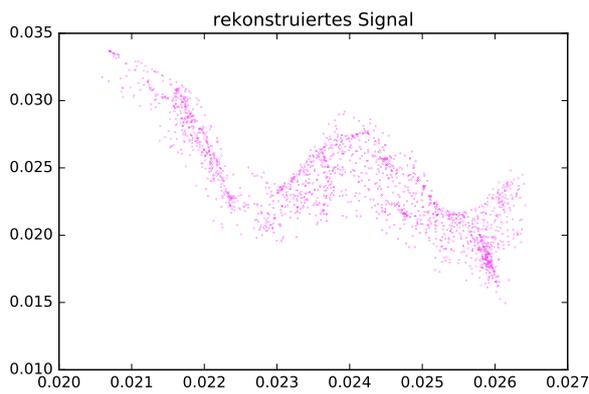
(b) 10 Durchläufe



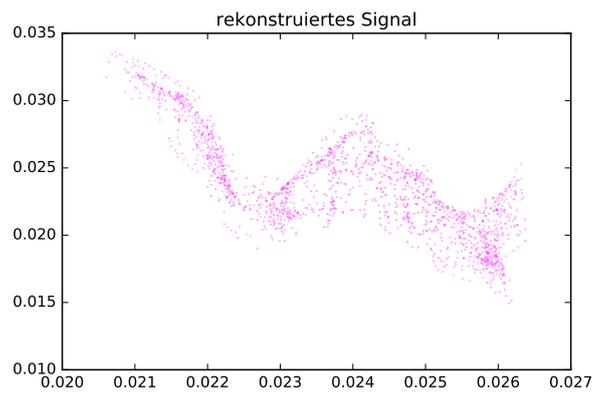
(c) 15 Durchläufe



(d) 30 Durchläufe

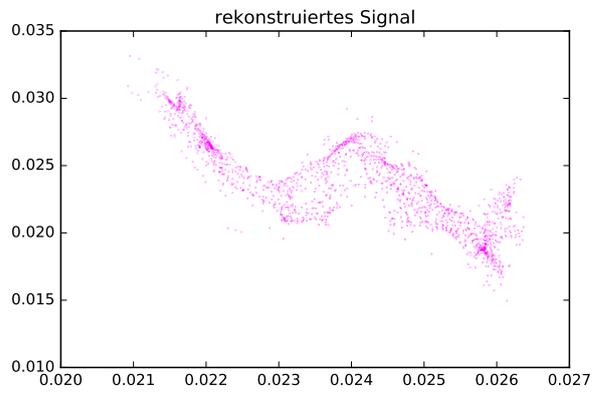


(e) 40 Durchläufe

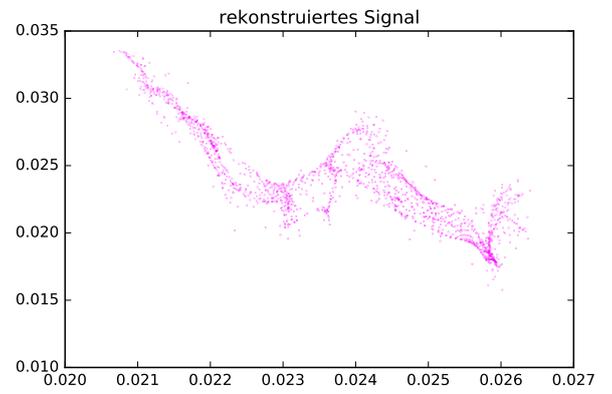


(f) 50 Durchläufe

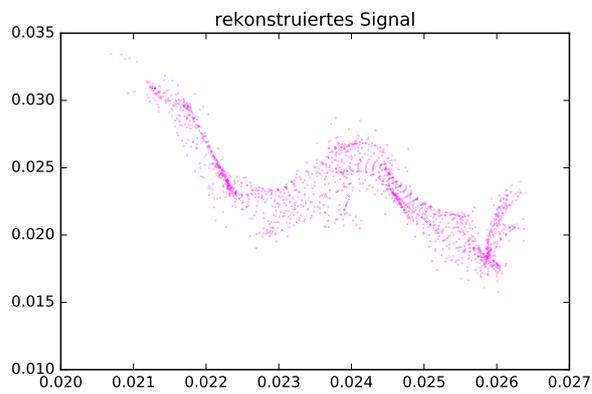
Abbildung 4.10: Verschiedene Durchlaufszahlen. Eine leichte Verbesserung ist erkennbar.



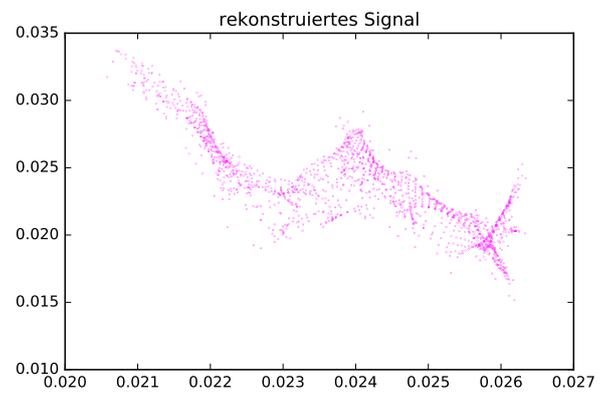
(a) $eps = 1,0$



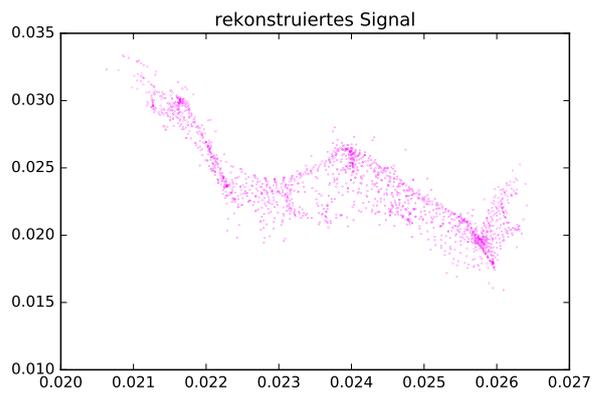
(b) $eps = 10^{-4}$



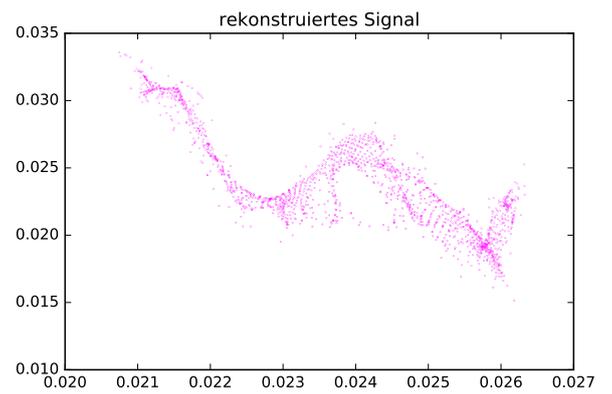
(c) $eps = 10^{-6}$



(d) $eps = 10^{-8}$



(e) $eps = 10^{-12}$



(f) $eps = 10^{-14}$

Abbildung 4.11: Verschiedene Toleranzniveaus. Ein direkter Einfluss ist dabei nicht zu erkennen.

5 Abschließende Betrachtungen

In der vorliegenden Arbeit beschäftigte ich mich mit der Untersuchung eines Verfahrens zum Abtasten von k -bandlimitierten Signalen und deren Rekonstruktion auf endlichen Graphen. Zum besseren Verständnis des Verfahrens stellte ich zuerst die Grundlagen des Gebietes *Compressive Sensing* und der *Signalverarbeitung auf Graphen* dar. Die Laplace-Matrix und deren Zerlegung $L = U\Lambda U^T$ wurde eingeführt und mit deren Hilfe die *Fourier-Transformation*. Des Weiteren wurde das Abtasten und Rekonstruieren in den Kontext eingebettet und auf Matrixmultiplikationen zurückgeführt. Ferner wurde die Fourier-Transformation genutzt um den Wirkungsbereich eines Filters auf den gesamten Graphen auszuweiten. Der so konstruierte Filter wurde durch ein Polynom approximiert. Im zweiten Teil beschäftigte ich mich genauer mit dem in [1] eingeführten Algorithmus: Zuerst wurde das Abtasten genauer beschrieben, welches randomisiert durchgeführt werden sollte. Dabei wurde, mit Respekt zu einer gegebenen Verteilung \mathbf{p} eine zufällige Indexmenge den Knoten entnommen. Die Abtastmatrix wurde dann mit Hilfe dieser Indizes erzeugt.

Die Theoretische Grundlage hierfür bildete erneut die Laplace-Matrix $L = U\Lambda U^T$ und die Kohärenz $v_{\mathbf{p}}^k$. Die Kohärenz diente dabei als Maß für die Güte einer gegebenen Wahrscheinlichkeitsverteilung \mathbf{p} im Vergleich zur Basis U . Darauf aufbauend wurde eine Verteilung \mathbf{p}^* auf den Knoten des Graphen hergeleitet. Sie ist optimal in dem Sinne, dass die entsprechende Kohärenz $v_{\mathbf{p}^*}^k$ den kleinst-möglichen Wert \sqrt{k} annimmt.

Im zweiten Schritt konnte ich darstellen wie die optimale Verteilung geschätzt werden kann, ohne die Zerlegung von L zu berechnen. Zu diesem Zweck wurden zufällige gaussverteilte Signale gefiltert, komponentenweise addiert und anschließend gemittelt. Dafür wurde ein polynomieller Filter genutzt.

Als Resultat der obigen Überlegungen wurde Algorithmus 1 angegeben.

Der obige Algorithmus wurde durch ein geeignetes Verfahren zur Rekonstruktion abgerundet. Ausgehend von einem ersten intuitiven Ansatz wurde das rekonstruierte Signal als Lösung eines Optimierungsproblems gegeben. Die zulässigen Lösungen waren dabei nur k -bandlimitierte Signale, was sich als nachteilig herausstellte. Daher wurde ein Regularisierungsterm eingeführt, der bandlimitierte Signale weniger bestraft als nicht-bandlimitierte.

Als Ergebnis besteht die Rekonstruktion im Lösen eines konvexen quadratischen Optimierungsproblems für das beliebige Vektoren als Lösung zugelassen sind. Die Eigentliche Rekonstruktion besteht somit im lösen eines linearen Gleichungssystems.

Die erste Reihe numerischer Tests untersuchte die synthetischen Datensätze *Bunny* und

Minnesota. Hierbei bestand das erste Vorgehen darin die numerischen Ergebnisse der Autoren zu reproduzieren. Nach großen Unklarheiten nahm ich Kontakt zu den Autoren auf, die mir freundlicherweise ihren MATLAB-Code zur Verfügung stellten. Damit konnten meine offenen Fragen geklärt und die Implementierung von Algorithmus 1 in PYTHON abgeschlossen werden. Auch das Verfahren zur Rekonstruktion konnte ich umsetzen. Durch die Reproduktion der Plots konnte ich ferner die Funktionalität meiner Implementierung beweisen.

Als nächstes untersuchte ich den Einfluss den die Parameter auf das Verhalten der Implementierung nehmen. Hier konnte ich zwei Bemerkungen der Autoren bestätigen und genauer studieren.

Erstens konnte ich durch meine Versuche falsifizieren, dass die Approximation an \mathbf{p}^* verbessert werden kann, indem Algorithmus 1 mehrmals mit den selben Parametern gestartet wird und die Summe der Ergebnisse gemittelt wird. Ergänzend konnte ich hier hinzufügen, dass die Approximationsgüte nicht beliebig verbessert werden kann. Gleichwohl kann hierdurch eine deutliche Verbesserung erzielt werden.

Zweitens konnte ich bestätigen, dass durch eine Regularisierung der Form $\gamma z^\top L^p z$ der Rekonstruktionsfehler deutlich verbessert werden kann, mittels einer entsprechenden Wahl von γ und p . Vor allem bei den zufällig erzeugten Signalen konnten sehr gute Ergebnisse erzielt werden. Die Parameter selbst können durch einen entsprechenden Plot leicht bestimmt werden.

Während der beiden obigen Beobachtungen konnte ich den Einfluss des Toleranzniveaus genauer untersuchen. Durch seine Auswirkung nur auf bestimmte Einträge der geschätzten Verteilung spielt dieser Parameter kaum eine Rolle bei der Betrachtung der Fehler. Allerdings kann sein Einfluss visualisiert und somit verdeutlicht werden.

Des Weiteren wurde deutlich, dass der Wert k weniger als wandelbarer Parameter gesehen werden sollte, sondern als fester Wert.

Schließlich konnte ich festhalten, dass die Anwendung der Methoden auf k -bandlimitierte Signal sehr vielversprechende Ergebnisse lieferte.

Im letzten Teil der Arbeit betrachtete ich Daten aus der Automobilbranche und versuchte zu verifizieren, ob Algorithmus 1 gewinnbringend angewendet werden kann.

Zuerst wiederholte ich die Versuche, die auf den synthetischen Datensätzen vorgenommen wurden. Der relative und der absolute Fehler zeigten ähnliche Verhalten und auch die Rekonstruktion der zufälligen Signale fügte sich in den Kontext der früheren Versuche.

Durch die Interpretation der x -Koordinaten als Signal auf dem Graphen konnte ich eine Abtastung vornehmen und die Rekonstruktion genauer prüfen. Analog tat ich dies für die y - und z -Koordinaten. Sodann bestimmte ich die Regularisierungsparameter. Ab diesen Punkt fügte sich die Untersuchung nicht länger in den vorherigen Kontext.

Die Fehlerplots zeigten ein viel schlechteres Verhalten und legten eine Potenz von $p = 1$ nahe. Durch kleinere Testreihen konnte ich die Güte der Approximation einschätzen und verschiedene Werte für k betrachten. Dabei konnte ich keinen geeigneten Wert für

k ausmachen. Des Weiteren konnte ich festhalten, dass selbst bei einer hohen Anzahl von Messungen die Rekonstruktion als nicht gut genug bezeichnet werden muss. Die Verbesserung der Rekonstruktion, die durch eine Wahl von einem hohen Werte für m erzielt wird rechtfertigt den Mehraufwand nicht.

Zusammenfassend kann ich zwei Dinge festhalten.

Zum ersten konnte ich keinen überzeugenden Wert für k ausmachen und folglich die Daten nicht als k -bandlimitiert einstufen. Die Ergebnisse der Rekonstruktion konnten im Hinblick als Anwendung zur Komprimierung nur wenig überzeugen.

Zur Bewertung der obigen Ergebnisse bleibt festzuhalten, dass Algorithmus 1 sehr gute Ergebnisse liefert, wenn der Wert für k genau feststeht. Hier konnte ich die Kernaussagen der Autoren bestätigen.

Gleichsam ist es fragwürdig, ob die unbefriedigenden Ergebnisse auf den industriellen Daten als Schwachstelle der Methode angesehen werden sollten oder eher als ungünstig gewähltes Einsatzgebiet. Die von mir erzielten Ergebnisse legen eher die Notwendigkeit einer Methode nahe, die entscheiden kann, ob ein Signal als k -bandlimitiert betrachtet werden kann oder nicht.

Notations- und Abkürzungsverzeichnis

Notationen

\mathbb{N}	Die Menge der natürlichen Zahlen: $\{0, 1, 2, 3, \dots\}$
\mathbb{R}	Die Menge der reellen Zahlen
$G = (\mathcal{V}, \mathcal{E}, \mathcal{W})$	Ein endlicher Graph mit Knotenmenge \mathcal{V} , Kantenmenge \mathcal{E} und Adjazenzmatrix \mathcal{W}
$L = U\Lambda U^\top$	Die Laplace-Matrix eines Graphen
U_k	$(u_1, \dots, u_k) \in \mathbb{R}^{n \times k}$ die ersten k Fourier Koeffizienten.
\mathbf{s}	Ein Signal auf einem Graphen G
$\hat{\mathbf{s}}$	Das Signal \mathbf{s} in der Spektraldarstellung
\mathcal{S}_G	Die Knotendomäne des Graphen G
$\hat{\mathcal{S}}_G$	Die Spektraldomäne des Graphen G .
h_λ	Der ideale Tiefpassfilter mit Abschneidefrequenz λ
κ_λ	Eine polynomielle Approximation an den Filter h_λ
$\mathbf{s}_{\mathcal{F}}$	Das durch den Filter \mathcal{F} gefilterte Signal. Dabei kann \mathcal{F} ein beliebiger Filter sein
M	Matrix, die zum Abtasten der Signale benutzt wird
\mathbf{s}_M	Das durch M abgetastete Signal ($\mathbf{s}_M = M\mathbf{s}$)

Abkürzungen

FEM	Finite-Elemente Methode
GFT	Graph-Fourier Transformation
IGFT	Inverse Graph-Fourier Transformation

Abbildungsverzeichnis

- Abbildung 2.1 Beispiel eines Graphen G und eines k -bandlimitierten Signals auf G mit $k = 30$. Das Signal ist zufällig und normiert. Der Graph stammt aus der DAVIDSENSORNET aus der MATLAB Bibliothek GSP.
- Abbildung 4.1 Der Bunny- und der Minnesota-Graph. Beide aus der Bibliothek PYGSP
- Abbildung 4.2 Rekonstruktionsfehler eines normierten, k -bandlimitierten Signals mit $k = 10$. Dabei sind die Koeffizienten unabhängig und standard-normalverteilt. Die Anzahl der Messungen beträgt $m = 200$.
- Abbildung 4.3 Absoluter und relativer Fehler der Verteilung $\tilde{\mathbf{p}}$ zu \mathbf{p}^* , bei verschiedenen vielen Durchläufen
- Abbildung 4.4 Visualisierung der geschätzten Verteilung für $eps = 10^{-8}$ und einem Durchlauf und $k = 100$.
- Abbildung 4.5 Der betrachtete Chevrolet. Zur Darstellung wurde *Animator4* genutzt
- Abbildung 4.6 Darstellung des Bauteil 2000002 in Python
- Abbildung 4.7 Testreihe zur Prüfung von $k = 100$. Dabei sollte geprüft werden, inwiefern das Signal als 100-bandlimitiert bezeichnet werden kann
- Abbildung 4.8 Plot der Verteilungen. Die Parameter sind $k = 100$ und $eps = 0.01$. Die Graphen stammen aus der Bibliothek PYGSP.
- Abbildung 4.9 Rekonstruiertes Signal bei unterschiedlich vielen Abtastungen. Als Parameter wurde $eps = 10^{-4}$, $k = 100$ und 1 Durchlauf gewählt.
- Abbildung 4.10 Rekonstruiertes Signal bei unterschiedlich vielen Durchläufen mit $eps = 10^{-4}$ und $k = 100$. Es wurden je 500 Messungen vorgenommen.
- Abbildung 4.11 Einfluss des Toleranzniveaus auf die Rekonstruktion. Hier ist eine Testreihe mit $k = 10$, $m = 200$ und je 15 Durchläufen gezeigt.

Literaturverzeichnis

- [1] G. Puy, N. Tremblay, R. Gribonval, P. Vandergheynst: *Random Sampling of Bandlimited Signals on Graphs*, NIPS2015 Workshop on Multiresolution Methods for Large Scale Learning, Montréal, 2015.
- [2] S. Chen, R. Varma, A. Sandryhaila, J. Kovacevic: *Discrete Signal Processing on Graphs: Sampling Theory*, Signal Processing, IEEE Transactions on 99, 2015.
- [3] A. Sandryhaila, J. M. F. Moura: *Discrete Signal Processing on Graphs*, IEEE Transactions on Signal Processing, vol. 61:(7), pp.: 1644-1656, April 2013
- [4] N. Perraudin, J. Paratte, D. Shuman, L. Maartin, V. Kalfolias, P. Vandergheynst, D. Hammond: *GSPBOX: A toolbox for signal processing on graphs*, <http://arxiv.org/pdf/1408.5781v2.pdf>, aufgerufen am: 07.04.2015.
- [5] D. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst: *The Emerging Field of Signal Processing on Graphs*, Signal Processing Magazine, IEEE, vol. 30, Nr. 3, S. 83-98, 2015
- [6] E. D. Napoli, E. Polizzi, Y. Saad: *Efficient estimation of eigenvalue counts in an interval*, <http://arxiv.org/pdf/1308.4275v2.pdf>, aufgerufen am: 02.05.2016
- [7] N. Tremblay, G. Puy, R. Gribonval, P. Vandergheynst: *Compressive Spectral Clustering*, 33rd International Conference on Machine Learning (ICML), New York, USA, 2016
- [8] M. Griebel, H.-J. Bungartz, C. Czado, J. Garcke, U. Trottenberg, C.-A. Thole, B. Bohn, R. Iza-Teran, A. Paprotny, B. Peherstorfer, and U. Schepsmeier: *SIMDATA-NL - Nichtlineare Charakterisierung und Analyse von FEM-Simulationsergebnissen für Autobauteile und Crash-Tests. Abschlussbericht des BMBF-Projekts*, 2014.
- [9] J. Garcke, R. Iza-Teran: *Maschinelle Lernverfahren zur effizienten und interaktiven Auswertung großer Mengen von CAE-Modellvarianten*, In: VDI-Tagung SIMVEC, 2014.
- [10] M. Fornasier, H. Rauhut: *Compressive Sensing*, In: Otmar Scherzer (Hg.): Handbook of Mathematical Methods in Imaging, S. 187-228, New York 2015.
- [11] P. Deuffhard, A. Hohmann: *Numerische Mathematik 1. Eine algorithmisch orientierte Einführung*, Berlin 2008.

- [12] J. Garcke, R. Iza Teran: *Data Analytics for Simulation Repositories in Industry*, In: E. Plödereeder, L. Grunke, E. Schneider, D. Ull (Hg.): GI-Tagungsband der Informatik 2014, S. 161-167, 2014
- [13] J. Garcke: *Scientific Computing II. Skript zur Vorlesung*, Sommersemester 2016.
- [14] B. Bohn, J. Garcke, R. Iza-Teran, A. Paprotny, B. Peherstorfer, U. Schepsmeier, and C.-A. Thole: *Analysis of Car Crash Simulation Data with Nonlinear Machine Learning Methods*, In: *Procedia Computer Science, Proceedings of the ICCS 2013, Barcelona, Serie 18*, S. 621-630 2013.
- [15] A. Sandryhaila, J. M. F. Moura: *Discrete Signal Processing on Graphs: Frequency Analysis*, *IEEE Transactions on Signal Processing*, June 30, 2013.
- [16] *The Graph Signal Processing Toolbox*, MATLAB-Version: <https://lts2.epfl.ch/gsp/>. Zuletzt aufgerufen am 29.08.2016.
- [17] *The Python Graph Signal Processing Toolbox* PYTHON-Version: <http://pygsp.readthedocs.io/en/latest/>
- [18] M. Ulbricht, S. Ulbricht: *Nichtlineare Optimierung*, Springer, 2012.
- [19] G. Puy, N. Tremblay, R. Gribonval and P. Vandergheynst: *Graph Sampling Box*, Veröffentlichter Code der Autoren von [1], <http://grsamplingbox.gforge.inria.fr/>. Zuletzt aufgerufen am 30.8.2016.
- [20] *The Numpy Package*: <http://www.numpy.org/>. Zuletzt aufgerufen am 31.08.2016.
- [21] *Dokumentation der scipy-Methode eigsh*, <http://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.sparse.linalg.eigsh.html>. Zuletzt aufgerufen am 31.08.2016.
- [22] Finite Element Model Archive: <http://www.ncac.gwu.edu/vml/models.htm>.